



# International Journal of Allied Practice, Research and Review

Website: [www.ijaprr.com](http://www.ijaprr.com) (ISSN 2350-1294)

## A Comparative Study of Software Testing Techniques Viz. White Box Testing Black Box Testing and Grey Box Testing

\*Taraq Hussain,\*\* Dr.Satyaveer Singh

\*Lecturer in Computer Science GDC Poonch

\*\*Assistant Professor Seth Motilal PG College Jhunjhunu

**Abstract:-**Testing can never completely identify all the defects within software. Instead, it furnishes a reproach or evaluation that compares the state and behaviour of the product against principles or mechanisms by which someone might recognize a problem. These predictions may include specifications, conventions' comparable products, past versions of the same product, inferences about intended or expected purpose, user or customer expectations, relevant standards, applicable laws, or other criteria. Software testing is important term for the reliability of software. Testing provides the genuine structure and validity to the software for the efficient performance in the operational condition. In this paper an outline of the software testing is being provided and emphasis is also given for its role in the reliability of software with comparative effort.

**Keywords:-**Software testing, Black box testing, White box testing Grey box testing.

### 1. Introduction

Programming testing is an investigation led to recognize absconds and give financial specialists data about the nature of a product item under test. It predominantly fills two needs:-

- Identification of defects; and
- Evaluating the quality of a product.

For deformity recognizable proof reason, programming testing is actualized beginning from necessity stage and completely through the advancement process until the last programming item is finished. Testing is a bit much the executing of system or application, yet can be executed on right on time break deliverables of an advancement process. Audit and examination (generally called static testing or static investigation) are regular intends to recognize absconds in framework examination and outline stage. There are static trying methods to audit project codes without really executing them. The destinations of these static examination systems are to recognize deformities at right on time phase of the advancement process, and find absconds that are not effectively uncovered through project execution. Testing through system execution (additionally called element testing) is typically executed at later phase of the advancement process when part or the majority of the system codes are

delivered. It looks at the physical reaction from the framework to variables that are not steady and change with time. Predefined test information are inputted to the framework to check if the yield is obviously. Element testing systems are powerful approaches to survey if the conveyed programming meets the indicated necessities and unspecified prerequisites.

Testing, normally dynamic testing likewise fills the need to assess the nature of last items. It gives a goal and autonomous perspective of a product item for the business to acknowledge and comprehend the danger of programming usage.

## **II. Goals and objective of testing**

There are four main objectives of testing

1. Evaluation: it assesses works under uncommon conditions and demonstrates that antiquities are prepared for fuse.
2. Revealing: it finds deformities, blunders and lacks. It administers framework capacities and impediments, nature of segments, work curios and the framework.
3. Anticipation: it gives data to avert or consolidate the quantity of slips, illuminate framework particulars and execution. Distinguish approaches to avoid hazard and issues later on.
4. Improving quality: by doing compelling testing we can minimize slips and consequently enhance the nature of programming.

A basic role for testing is to distinguish programming disappointments with the goal that the deformities may be revealed and adjusted. Testing can't build that an item capacity legitimately under all conditions however can just secure that it doesn't work appropriately under particular conditions. The complete objective of the product testing is to finish the items for the viable utilization and proficient execution. It gives the forecast to the product to produce the quality and proficiency with in the particular time. The product testing break down the antiques for proficiency as well as raise the items for the future execution.

## **III. Black Box Testing**

BLACK BOX TESTING otherwise called behavioural testing, is a product testing strategy in which the interior structure/ outline/execution of the thing being tried is not known to the analyser. This strategy for endeavours to discover blunders in the accompanying classes

- Incorrect or missing capacities
- Interface blunders.
- Errors in information structures or outside database access
- Behaviour or execution blunders
- Initialization or execution blunders.

Black box testing treats the system as a "disclosure", so it doesn't explicitly use Knowledge of within structure or code. Then again toward the end of the day the Test expert require not know the internal satisfying desires of the "Black box" or Main concentrate in discovery testing is on usefulness of the framework in general. The term 'behavioural testing' is additionally utilized for discovery testing and white box testing is likewise now and again called 'structural testing'. Behavioural test configuration is marginally not the same as discovery test outline on the grounds that the utilization of inward information isn't entirely illegal, yet's despite everything it debilitated.

Every testing system has its own particular favourable circumstances and disservices. There are a few bugs that can't be discovered utilizing just black box or just white box. Greater part of the

application is tried by discovery testing technique. We have to cover larger part of experiments so that the vast majority of the bugs will get found by discovery testing. Discovery testing happens all through the product improvement and testing life cycle i.e in Unit, Integration, System, Acceptance and relapse testing stages

### **3.1 The following testing techniques are used in black box testing**

- 1) **Equivalence Partitioning:** This procedure separates the information area of a system into comparability classes from which experiments can be inferred, so it can decrease the quantity of experiments.
- 2) **Boundary Value Analysis:** It concentrates on testing at limits, or where the amazing limit qualities are picked. It incorporates least, most extreme, only inside/outside limits, blunder qualities and normal qualities.
- 3) **Fuzzing:** This strategy sustains arbitrary information to application. It is utilized for discovering execution bugs, utilizing contorted/semi-distorted information infusion in a mechanized or semi-computerized session.
- 4) **Cause-Effect Graph:** In this strategy, testing starts by creating a diagram and making the connection in the middle of impact and its causes.
- 5) **Orthogonal Array Testing:** It can be connected where information area is little, yet excessively substantial to oblige comprehensive testing.
- 6) **All Pair Testing:** In this strategy, experiments are intended to execute all conceivable discrete blends of every pair of information parameters. Its primary target is to have a situated of experiments that covers all the sets.

### **3.2 Advantages:**

1. Analysers require not to have learning on particular programming dialect.
2. Testing is carried out from client's perspective.
3. It serves to uncover any ambiguities or irregularities in the prerequisite particulars
4. Developer and analyser both are autonomous of one another.

### **3.3 Disadvantages:**

1. Experiments are difficult to plan without clear details.
2. Possibilities of having redundancy of tests that are now done by software engineer.
3. A few sections of back end are not tried whatsoever.

## **IV. White Box Testing**

**White Box testing** is a product testing strategy in which the inner structure/outline/usage of the thing being tried is known to the analyser. The analyser picks inputs to practice ways through the code and decides the fitting yields. Programming skill and the execution learning is key. White box testing will be trying past the client interface and into the bare essential of a framework.

### **4.1 The following testing techniques are used in white box testing**

- 1) **Desk Checking:** Desk checking is the essential testing done on the code. The creators who have learning in the programming dialect extremely well will be included in work area checking testing.
- 2) **Code Walkthrough:** In this testing process a gathering of specialized individuals experience the code. This is one kind of semi-formal survey procedure.
- 3) **Formal Inspections:** Inspection is a formal, proficient and efficient strategy for discovering slips in outline and code. It's a formal audit and went for distinguishing all issues, infringement and other reactions.
- 4) **Control Flow Testing:** It is a structural testing methodology that uses the project control stream as a model control stream and supports all the more yet less difficult ways over less however convoluted way.
- 5) **Basis Path Testing:** Basis way testing permits the experiment creator to create a coherent intricacy measure of procedural outline and after that uses this measure as a methodology for illustrating a fundamental set of execution ways.
- 6) **Data Flow testing:** In this sort of testing the control stream diagram is clarified with the data about how the system variables are characterize and utilized.
- 7) **Loop Testing:** It only spotlights on the legitimacy of circle develop.

## **V. Grey Box Testing**

Grey Box Testing is a product testing system which is a blend of Black Box Testing strategy and White Box Testing technique. In Black Box Testing, the inward structure of the thing being tried is obscure to the analyzer and in White Box Testing the inner structure is known. In Grey Box Testing, the inside structure is somewhat known. This involves having access to internal data structures and algorithms for purposes of designing the test cases, but testing at the user, or black-box level.

### **5.1 The following testing techniques are used in Grey box testing**

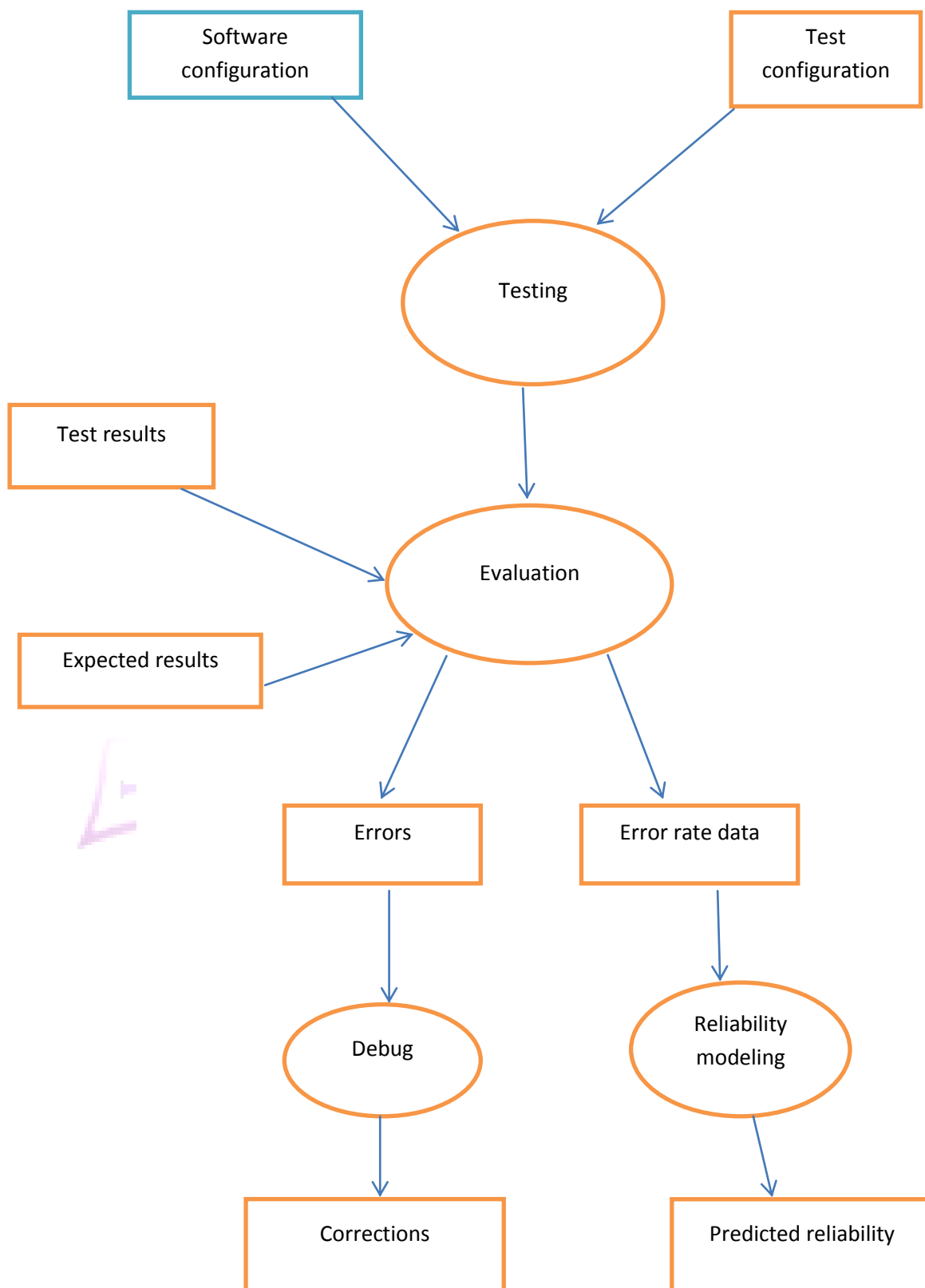
- 1) Orthogonal Array Testing: This kind of testing use as subset of all conceivable blends.
- 2) Matrix Testing: In lattice testing the status report of the task is expressed.
- 3) Regression Testing: If new changes are made in programming, relapse testing suggests running of experiments.
- 4) Pattern Testing: Pattern testing checks the great application for its structural engineering and configuration.

### **5.2 Advantages:**

1. It gives consolidated profit of discovery and white box testing procedures.
2. In dim box testing, analyzer can plan incredible test situations.
3. Impartial testing
4. Make a savvy test composing.

### **5.3 Disadvantages:**

1. Test scope is restricted as the entrance to source code is not accessible.
2. Numerous system ways stay untested.
3. The experiments can be repetitive.



**Fig 1. Test information flow**

S.No.	Black box testing	Grey box testing	White box testing
1.	Analyses fundamental aspects only i.e no knowledge of internal working	Partial knowledge of internal working	Full knowledge of internal testing
2.	It is least exhaustive and time consuming	It is somewhere between the two	Potentially most exhaustive and time consuming
3.	Granularity is low	Granularity is medium	Granularity is high
4.	Performed by the end user and also tester	Performed by end user and also tester	It is performed by developer and tester
5.	Testing is based on external exceptions, internal behaviour of program is ignored	Test design is based on high level database diagrams, data flow diagrams, internal state knowledge of algorithm and architecture	Internal are fully known.
6	It can test only by hit and trial	Data domains and internal boundaries can be tested and overflow if known	Test better, data domains and internal boundaries
7.	No suited for algorithm testing	No suited for algorithm testing	It is suited for algorithm testing.

**Table 1. Comparison between three types testing**

Table 1. Show the comparative anatomy of the three testing techniques. Although the tester cannot implement all the testing technique in assemble and produce the direct effect to the project. Since the time and cost are the major constraints for the experimental testing? We comparatively analyse the three testing techniques and on the experimental study, it has found that the white box testing techniques ensure minimum defects and which are unable to cause any failure to the concerning project under analyses.

## **VI. Software Reliability**

Programming testing and programming dependability have generally fit in with two different communities. However at present there is an in number security in the middle of software testing and programming unwavering quality. An imperative part of testing is to make quality and its qualities unmistakable which incorporate the dependability of the product. The unwavering quality characteristic is not directly measurable and should along these lines be gotten from different estimations, for example, disappointment information gathered amid testing. Software testing is a viable system for evaluating the present dependability and foreseeing future dependability furthermore to improve it. The trouble of the dependability quality is that it only has an importance on the off chance that it is identified with a particular client of the framework. Diverse clients experience distinctive unwavering quality, in light of the fact that they utilize the framework as a part of diverse ways. In the event that we are to gauge, foresee or affirm the unwavering quality, we must relate this to the use of the framework. Restricted of relating the dependability to the use is to apply utilization based testing

Then again programming dependability can be utilized to gauge the amount of advancement has been made in framework level testing. The measure of an upkeep work can be dictated by the measure of framework dependability that can be yielded for some time. The idea of unwavering quality likewise permits us to measure the disappointment related quality part of a product framework. Evaluation of the quality part of programming frameworks gives designers and chiefs a superior

understanding into the procedure of programming advancement. In future, it is critical to bring these two gatherings all the more nearly, so that on one hand, programming testing can be adequately directed, while then again, programming unwavering quality can be precisely measured and moved forward.

On experimental study, when we compare the three testing techniques, we found that the White box testing technique analyse and detect the defects, similarly remove the defects more profoundly and accurately, if the some of the defects remain, they have negligible effects for the failure of the concerning the project that have taken under consideration.

On experimental study we have deduce the following results of comparative reliability of the three testing techniques.

Software Testing Techniques	Defects Detection	Effect on the overall failure	Reliability
White Box Testing	73%	22%	78%
Grey Box Testing	63%	32%	56%
Black Box Testing	42%	39%	32%

Table 2.

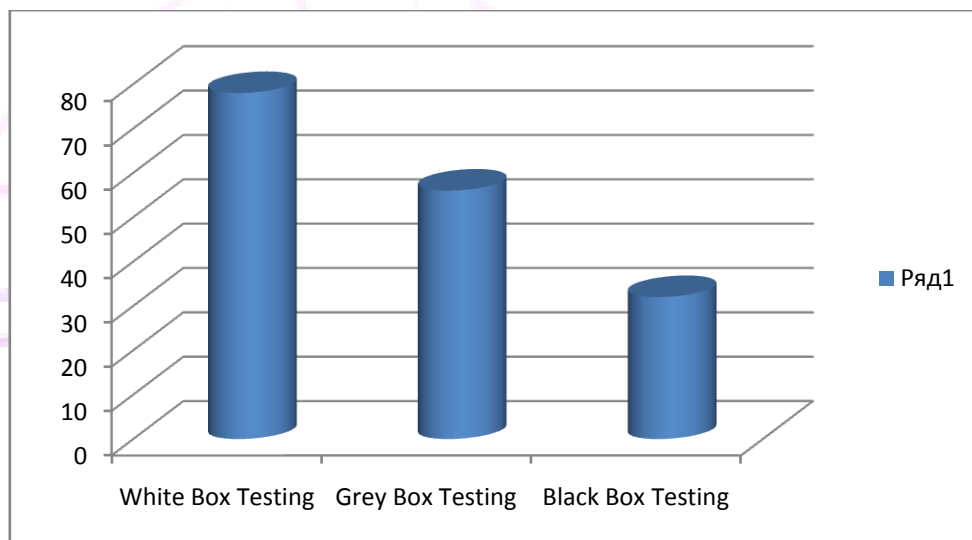


Fig.2 Comparative Reliability of the three testing techniques.

## VII. Conclusion

Software Reliability is the product of the accurate testing techniques and a in depth analysis of the above said testing techniques will have profound effect on the accuracy and tolerability of the software. On Comparison of the three testing techniques viz. White box, Grey Box, and Black box, we found that White box testing techniques produce better results for the reliability of the software.

## VIII. References

- [1] NIST-Final Report “The Economic Impacts of InadequateInfrastructure for Software Testing”, *Table 8-1*, NationalInstitute of Standards and Technology, May 2002.
- [2] Bertolino, A. (2007). , “Software testing research: Achievements, challenges, dreams.”, *In FOSE '07: 2007Future of Software Engineering*, pages 85–103, Washington,DC, USA. IEEE Computer Society.

- [3] C. Wohlin, M. Höst, P. Runeson and A. Wesslén, "Software Reliability", in Encyclopedia of Physical Sciences and Technology (third edition), Vol. 15, *Academic Press*, 2001
- [4] S. Dalal and C. Mallows, "When Should One Stop Testing Software", *Journal of the American Statistical Associations*, Vol. 81, 1988, pp. 872–879.
- [5] M. C. K. Yang and A. Chao, "Reliability-Estimation and Stopping-Rules for Software Testing, Based on Repeated Appearances of Bugs", *IEEE Transactions on Reliability*, June 1995, pp. 315–321.
- [6] Kshirasagar Naik, Priyadarshi Tripathy, "Software testing and quality assurance: theory and practice", *John Wiley & Sons*, 2008.
- [7] M. R. Lyu, "Software Reliability Eng : A Roadmap," *Intl. Conf. on Software Eng. (FOSE '07)*, May 2007, pp. 153-170.
- [8] Sheikh Umar Farooq and S.M.K. Quadri, "Effectiveness of Software Testing Techniques on a Measurement Scale", *Oriental Journal of Computer Science & Technology*, Vol.3(1), 109-113 (2010).
- [9] P. Frankl, R. Hamlet, B. Littlewood, and L. Strigini. Choosing a Testing Method to Deliver Reliability. In *International Conference on Software Engineering*, pages 68–78, 1997.
- [10] S.M.K Quadri and Sheikh Umar Farooq, "Software Testing –Goals, Principles, and Limitations", *International Journal of Computer Applications* (0975 –8887) Volume 6– No.9, September 2010.
- [11] S. M. K. Quadri and Sheikh Umar Farooq, "Testing Techniques Selection: A Systematic Approach", *Proceedings of the 5th National Conference; INDIACOM-2011*, pp-279-281, March 10 – 11, 2011.
- [12] Sheikh Umar Farooq, and SMK Quadri, "Identifying some problems with selection of software testing techniques", *Oriental Journal of Computer Science & Technology* Vol.3(2), 266-269 (2010).
- [13] Software Testing. Gregory M. Kapfhammer. The Computer Science and Engineering Handbook, CRC Press. May, 2004.
- [14] Mohd. Ehmer Khan, "Different Approaches to Black Box Testing Technique for Finding Errors," *IJSEA*, Vol. 2, No. 4, pp 31-40, October 2011
- [15] Jovanovi?, I. "Software Testing Methods and Techniques. " *The IPSI BgD Transactions on Internet Research*: 30
- [16] F. Saglietti, N. Oster, and F. Pinte, "White and grey-box verification and validation approaches for safety- and security-critical software systems," *Information Security Technical Report*, vol. 13, no. 1, pp. 10–16, 2008.
- [17] T. Murnane and K. Reed, "On the effectiveness of mutation analysis as a black box testing technique," in *Software Engineering Conference*, 2001. *Proceedings. 2001 Australian*, 2001, pp. 12 –20.
- [18] Nidhra, Srinivas, and Jagruthi Dondeti. "Black box and white box testing techniques- A Literature." *International Journal of Embedded Systems & Applications* 2.2 (2012).
- [19] Khan, Mohd Ehmer, and Farmeena Khan. "A Comparative Study of White Box, Black Box and Grey Box Testing Techniques." *International Journal of Advanced Computer Sciences and Applications* 3, no. 6 (2012): 12-15
- [20] Grey Box Testing from Wikipedia available at [http://en.wikipedia.org/wiki/Gray\\_box\\_testing](http://en.wikipedia.org/wiki/Gray_box_testing)
- [21] P. Mitra, S. Chatterjee, and N. Ali, "Graphical analysis of MC/DC using automated software testing," in *Electronics Computer Technology (ICECT)*, 2011 3rd International Conference on, 2011, vol. 3, pp. 145 –149.