A conceptual study of Test case design by Investigating the various principles and aspects.

Pramod P. Jadhav[1] , Dr. Shashank D. Joshi [2] , Dr. Anant M.Bagade[3]

1. Research scholar , Bharati vidyapeeth dhanakawadi-43,pune,Maharashtra, India.

2. Professor in Bharati vidyapeeth pune, Dept. Of. Comp. Engg. Maharashtra, India.

3. Associat Professor in PICT, Dept. of IT, pune-43, Maharashtra, India.

Abstract : *Test case is a important factor in the testing. It is prefers in concept test driven development (TDD). The main background of this concept were using/writing the test case,because many things are taken into consideration while writing the test case.The aim of this paper is apply the test case principles and its aspect properly. Otherwise it directly affects on testing and ultimately on project results. Test case is playing an important role in the testing phase. Test case should be written in either manually or automated. In the market there are ready made tools are available. Like test director. In the traditional testing, test case is manually generated. The measure part of this paper are to discuss various test case aspect, and its principles. A tipical test bed formate is also mention for reference.*

Keywords : Test, Test case, TDD, Model Transformation.

I. INTRODUCTION

Test case is important because through that developer can understand that how much your project is efficient and to make more effective what precautions is to be taken, in testing and test case point of view. Test case is work as a component which checks whether the input, function, and overall process of the project is working properly or not. In short the test case is the mirror of the project development. So it is an important phase in the software development life cycle (SDLC). In SDLC life cycle testing is having more wattage. While talking the test case generation method, it is having following field, test case id, unit to test, expected result, actual result, pass/fail, test data, executed test, and finally comment. Most of the time test, test case name, test case number, design by, executed by, design date, executed date, small description, step number, action, etc. is also mention in the field of test case. So these many filed are necessary, and should cover maximum out of it, while test case are concern.

Another side of the test case is it is used for well known concept test driven development/design (TDD) . In the test driven design(TDD) test case is the preferable task, and which is core part of TDD. Because TDD process is based on test case and other related concept [1][2][3].

Test case/TDD is later used for model driven development (MDD). Test case and MDD generally used for model transformation process. This process require different model and its related transformation tools.

A] Dependencies for Test Case:

Test case is depends on various factor like, main function, actual flow of development, current position of software project, etc. in test driven development test case is the prime issue. Because without writing test case TDD process will not start. To successful execution of TDD, test case is playing an important role in the TDD. To generate the test case there are advance tool are available in the market of software engineering. So TDD process is depend on test cases, and test case is depends on, above mention factor[4].Test can be derive in different specification or methods. The requirement base specification method include various form called boundary value analysis (BVA) partition equivalence (PE), state transition methodology,use case testing, etc. another method to derive the test case is the structure of system or structure of code, to get into analysis of code, developer must check the coverage of statement, path basis,and branch coverage, which helps to create the test case on proper formate. And finally test case should be drawn on the basis of testers experience. So these are the few criteria based specification to generate the test cases. on the basis of above discussion we can made a better quality of test case. There are few more option to improve the quality of test case and overall testing strategy [5][6].below are the principles which support the better quality of test case and testing.

II. PRINCIPLES OF TEST CASE

A] Boundary Value Analysis (BVA) : boundary value analysis is a testing technique, which define the boundary value of code. The main focus is on word boundary. The extreme input of the domain value , are give the maximum error for specific code. More application error are generate at the boundary of specific domain. So to identify and overcome the error of these system, boundary value analysis is play an important role in the domain of testing. Boundary value analysis is the later stage of partition equivalence. BVA technique is helpful in big scale as well as complicated project. So as the project will develop, complication and maintenance will be overcome.

B] Partition Equivalence(PE) : The require domain data value are divided in to the same or equivalence classes are called a partition equivalence(PE). One test value is select from a single class for testing. So the total test case are reduce. This is one of the method to concise the test case formate. Because number of test case is increased , maintenance of the test case are also increased. To maintain the test case and its maintenance , boundary value analysis and partition equivalence , is the best method of testing. Below figure 1,show the partition equivalence(PE), which show a main domain,which categorize with three classes.

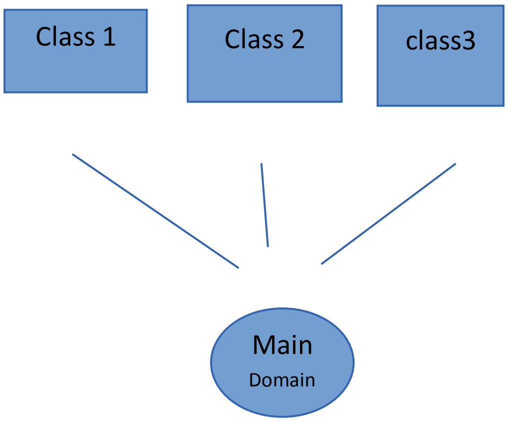

Fig 1: partition equivalence(PE)

C] State Transition Methodology : in state transition method, different state of case is initialized. The output state are change depend on the input state. The flow of state is to be maintain according to the flow of system. Some time state may be pass or fail, it is up to the current situation of the project. In this method sequence of test case is play an important role in the overall project. If sequence is not follow the result of the system is may affect. So state of test, its transition, its sequence, its result, and input/output condition , all these factor must be traced properly while using state transition method.

D] Use case testing : Use case is a component which execute the query step by step. In use case testing every case is modify before or after the execution. According to the project flow the case may be append . The structure or flow of the code may matter when the use case is written. Use case testing is focus on end user. Based on the structure, behavior,view of the user, use case may vary. And based on the same factor input output scenario is also change. So use case testing and its strategy is pointing out the overall behavior of the system[10].

E] Test bed: is the platform on which the project or software will test on the specific environment. e.g. which Operating system is require, what is the network platform,if necessary , detail of network structure, which browser is most suitable, which software background is comfort, which hardware background is needed, version of hardware and software, detail of hardware or total configuration of computer, etc. this information is important to frame the test bed setup.

Typical Test Bed :

OS :	windows/ linux.
Browser :	IE/firefox/chrome.
Network :	Server- apache,
	Cable – cat 5,
	Topology – star/mesh/bus,
	Device – AP/Router/Switchs.
Software :	JAVA/ .NET
Version :	6.1 JAVA
Hardware :	HD- 500GB,
	RAM – 4 GB,
	processor – i3/i5.

Above example are clear the idea of whole software project, and its requirement. test bed gives us a detail of how and what kind of environment is frame for the experimental setup of given project. According to test bed set up,and given input data,developer may predict the expected result of the project. In support with test cases strategy of test driven development(TDD), and test bed analysis, developer /tester can make the project on the highest level of success. Above information is also important for finalizing the deadline of the project.

Before writing the test case few point take into consideration. To make a test effective it should be written in a proper formate.

Knowledge of the subject : at the time of framing the test case, developer should have a deep knowledge of specific subject. Developer or tester having a enough experience to creating test case. This experience is very much helpful for writing the test case. While writing the test case, supporting factor should be take into consider. Because if you have a sufficient knowledge of required subject it will help to make your test case more realistic and useful . Collect information as per your requirement. While writing the current test case be aware about the previous test case and also keep the plan of next test case. So by analyzing the both the case current situation or behavior will be handle.

writing language : while writing the test case keep tour language as simple as possible. Because as your language is simple the execution of the test case is also very fast and accurate. For getting the more accurate and concise test case, language of test case is formal and decent. Simple language save the time of execution, and give the result more correct. These and many more factors are directly affecting on the quality of test[7]. One more precaution should be taken while framing the test case, is delete the repeated data , and new form of information should be generate. Another side of coin is, merge the equivalent information, and get it correct immediately.

While writing the test case considers the supporting factor like plan of project, designing aspect, blue print, SRS, etc. Software requirement specification (SRS) plays an important role in the design of test case. SRS gives us more information related to the project, and its design. Be careful while writing the test case and follow the test case standard protocol, so that written test case is more effective and efficient.

F] Format Of Test Case : while talking about the format of the test case,   initial paragraph of this document we comment on the format and field/ factor of test case. Following are the Few field : test case id, unit to test, expected result, actual result, pass/fail, test data, executed test, and comment. In the market few tools are available for automatic generation of test case. Few test case are generated on model driven basis , and few are based on STB like tools. Automatic generation is a common strategy in the world software development [8][9].

1] Test case id : give or generate the proper id of test case, because id is helpful for tracing the test case count. Through the id we can find the require test randomly, and also trace the sequence of test case. Further it is important for generating the final report.

2] Test case name : every time provide the title of the test case. So that developer can easily identify, which test is executed. Along with the id, name of test is equally important. It directly show a what kind of test be executed in the process of testing.

3] Unit to test : this field focus, to which specific portion of the code is tested, and what is the resultant factor of that specific unit code.

4] Expected result : In this field , write predicted result after the execution of test case .this field may use for comparison of the expected and original result. After the comparison you will find out gap between two result and developer cultivate the programming factor according to the original result. So it is important for move the software development in proper direction.

5] Actual result : actual result show a original output of the program. This field tell us that where you are actually? ,whether your development is going in right direction or not. How much improvement is necessary for achieving a final goal? Other factor is also supporting to verify the original result. So this field is very important.

6] Pass/fail test case : by analyzing the current scenario of development, few test case are fail and few test case are pass. According to the status of pass/ fail, developer derives the test case data, and refactors the code accordingly. Refactoring code is one of the most important steps in the agile development.

7] Test data : test data indicate that which data is provided as a input data. Some time input data may change on the basis of programming structure. So input data may vary on current situation of software.

8] Comment/conclusion: this is the tab which give a overall summery of the test case. By filling the above all option we developer gives a comment on test case and its behavior. Which test is executed successfully, which test fails, what is the reason for passing or failing the test, all these information must be filled properly. So that by reading summery we reached some concrete conclusion. Which is helpful for the final report writing. this information provides a detail idea of test case, as well as overall idea of project.

III. ASPECTS OF TEST CASE DESIGN

A] Test plan : test plan is a important step in the whole testing process. test plan is also have some criteria to frame. While preparing the test plan few perspectives should be taken into account. Software risk issue, strategy for the testing, requirement analysis, approach for testing, criteria of pass/fail, supporting tools, responsibility, shipping criteria, test case, report generation, provide guidance, these are few of them. Figure 2 indicate common steps of Test plan. This is standard steps but according to project and test design it is slightly update. Test plan provide us a good platform for project. A good test plan save the time of overall project. And it will helpful for prepare the future plan of project. Most of the time tester or developer itself unaware of bug available in the software. But after successful execution of test plan will get the detail idea of defect and its solution. Without test plan it is highly impossible to manage the project time, quality, and beget. One more important point is model analysis. Because in test plan we are thinking of test condition, test case, its solution and many more point, so to become a project condition more concrete, analysis of domain and its detail study are necessary [11].

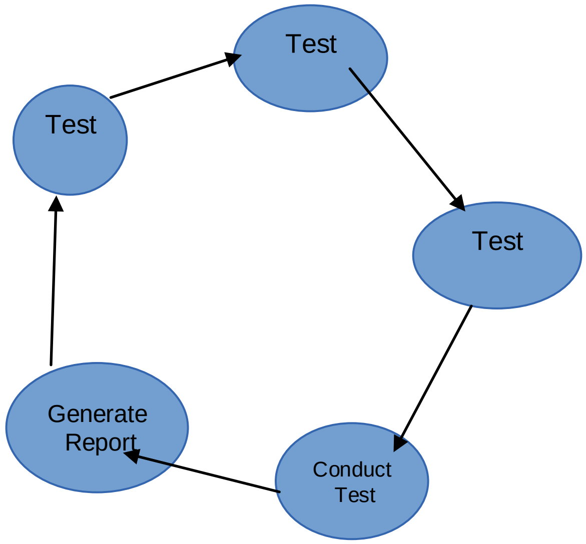

Fig 2: Common steps of Test plan

B] Software risk issue: software risk is major issue in the development process. While handling the risk in the project, first find out that which module of the project will test, how to test the specific part, which method will apply for testing, how much amount of time will require, and many more issue will be consider. So project team should solve the different risk factor very tactfully, and with concrete planning.

C] Strategy for the testing: before testing the software developer should decide the how to test the product, which method is apply for the testing. So decide the strategy behind the testing, and also find out the supporting tools , and raw material require for the testing. strategy of testing is play an important role for efficient and early testing. If strategy become powerful then final result also become a more correct and successful[10].

D] Requirement analysis: requirement analysis is initial and important step in every testing process. Requirements of project, create a base of software project. Which make them very strong and perfect towards the goal. Because of requirement and its proper analysis, project team frame a background out of it and it will helpful for further development of software. weak requirement analysis is becomes a time consuming process.

In the TDD, test case is a important requirement of TDD development. TDD require detail knowledge of test case and test generation. So all this analysis and overall test case concept are important for TDD[13][14].

E] Approach for testing:be very careful when finalize the testing approach. Because in testing world there are multiple methods are available. To select best one is a challenging task for tester/ developer. If developer select a wrong choice it become a hectic for further process. So decide a good testing approach, while testing. Even redeemed tool are available in the market, to select a proper tool, according to project requirement and be focus for smooth testing process. One more thing while selecting the testing approach is, do a proper study of project and testing method, find out the prose and cons to each and every methods, and then finalize the proper approach.

F] Criteria of pass/fail: most of the test case has based on the criteria pass and fail. Mainly in test driven development(TDD) flow of the testing is on, whether the test is fail or pass, depend upon the pass and fail result, direction of testing is finalize. So for correct direction decide the criteria of pass and fail. If developer decide the wrong criteria, the flow of the project may change. And finally it become a time consuming. This criteria may relevant to the final result of project. So deciding the criteria, consider the client suggestion,goal of the project, and many other perspective.

G] Supporting tools: in the world of testing, there are many more testing tols are available. Few tools are support very well. Most of the developer are taking support of this tools. Before using these tool make sure that which tool is most preferable, which tool is easy to use, which tools gives a more correct output, etc. these tools make testing process more efficient and effective. So select the tools which gives a more benefit, as concern to a successful project, and which having free or available in low cost.

H] Responsibility : responsibility in terms of project testing, responsibility in terms of shipping project, responsibility in terms of client feedback, and many more responsibility concern to project are take into consideration. While facing each and every responsibility, project team should be take care of all relative point of concern field. All this field, testing team having its won responsibility. Because in testing phase developer gown through a different steps. Each and every step is responsible for successful execution of every next phase. So developer/tester will be careful when the process is going on.

While handling all these responsibility a touch of extreme programming is also necessary, it teaches a lot of things regarding the responsibility,planning,and related process[12].

I] Shipping criteria : Team has to decide the shipping criteria, like when to deliver the product, how to deliver, what cases should be consider while delivering the product. Consideration of customer suggestion, how to deal with customer, what point should be discuss with customer, these and many more point should be take into consideration while shipping the project to the customer.

J] Test case: Basically a test case is an input data which help to develop the further code. Test case is the back bone of the testing. Test case predicts the expected output, and guide according to that. Test case should be a good quality. Good quality in the sense, it clear the meaning of case, it is that much transparent to develop the code very easily. Test case provides proper information towards the project. It should have a limited step. It may direct the flow of projects.

K] Report generation: report generation is a final step of the testing. This report should in a proper format; report should clear the idea of testing. It clears what has been tested. How it is tested , how much percent it find the bugs, what procedure is apply for the testing, resultant output is correct or not, if not what should be the next step, which method is integrated for testing, etc. Report gives an overall summery of the whole process. Use a proper format of report, which clear the idea of previous as well as future plan of project. Based on the report, developer can frame the future plan of the project, so this is very important aspect in testing and as well as overall development process.

This report generation process direct or indirect related to model transformation. Model transformation is a big process which include the multiple sub process. And combining them all, we get a stable transformation process[15].

IV. CONCLUSION

This paper is projects the different view of test case. And much more information related to test case. Along with this it show the basic principles & Different aspects of test case design. This paper is also focus on how to generate the test case with different specification. Paper starts from introduction of test case and ends with different aspect of test case design along with report generation. Maximum points are try to involve as a package of test case. Formate of test case is also mention with the support of strategical issue. while writing any test case what field are important as far as your/project requirement that are put up with example. Example of test bed setup is given in specified formate. All these stuff are important for efficient working of TDD, MDD, model transformation, and related concepts of software engineering.

V. REFERENCES

1) Pramod Jadhav ,Dr. Shashank Joshi.“Analysis of Test Driven Development For Software Design” international journal of computing and technology (IJCAT) , volume 1, Issue 6, ISSN : 2348-6090 , page no. :149-152 July 2014

2) Pramod Jadhav, Dr. Shashank Joshi "Survey on Test Driven Development for feasibility: An Industrial Approach " ,International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395 -0056 Volume: 02 Issue: 07 | Oct-2015 p-ISSN: 2395-0072, page no. 931 – 938

3) Adnan Cauevic Sasikumar Punnekkat Daniel Sundmark “Quality of Testing in Test Driven Development “ Lisbon, TBD, Portugal Portugal September 03-September 06 ISBN: 978-1-4673-2345-1 2012

4) Ratchanok Chaipraserth, Adisorn Leelasantitham,Supaporn Kiattisin “ A Test Automation Framework in POCT system using TDD Techniques” 2013 13th International Symposium on Communications and Information Technologies (ISCIT) 978-1-4673-5580-3/13/$31.00 © 2013 IEEE

5) Adnanˇ Cauˇ sevi´ c, Daniel Sundmark and Sasikumar Punnekkat Mälardalen University, Sweden“Test Case Quality in Test Driven Development: A Study Design and a Pilot Experiment” Proceedings of the EASE 2012 - Published by the IET ISBN 978-1-84919-541-6.

6) Dusica Marijan, Vladimir Zlokolica, Nikola Teslic, Tarkan Tekcan, and Vukota Pekovic “User-Driven Optimized Test Case Design and Modeling for End-User Device Quality Inspection “ , 2011 IEEE International Conference on Consumer Electronics (ICCE).

7) Adnan Causevic Daniel Sundmark “Factors Limiting Industrial Adoption of Test Driven Development: A Systematic Review “ Berlin, Germany March 21-March 25 ISBN: 978-0-7695-4342-0 2011

8) Yang Liu ,Yafen Li, Pu Wang “Design and Implementation of Automatic Generation of Test Cases Based on Model Driven Architecture “2010 Second International Conference on Information Technology and Computer Science.

9) Tarkan Tekcan, Vladimir Zlokolica, Vukota Pekovic, Nikola Teslic and Mustafa Gündüzalp “User-driven Automatic Test-case Generation for DTV/STB Reliable Functional Verification” 0098 3063/12/$20.00 © 2012 IEEE.

10) Jean-Jacques Gras, Rishabh Gupta, Elena Pérez-Miñana European Software and System

Engineering Research Lab Motorola Labs "Generating a Test Strategy with Bayesian Networks and Common Sense" Proceedings of the Testing: Academic & Industrial Conference – Practice And Research Techniques (TAIC PART'06) 0-7695-2672-1/06 $20.00 © 2006 IEEE.

11) Ruilian Zhao, Michael R. Lyu, Yinghua Min," A New Software Testing Approach Based on Domain Analysis of Specifications and Programs" Proceedings of the 14th International Symposium on Software Reliability Engineering (ISSRE'03) 1071-9458/03 $ 17.00 © 2003 IEEE

12) Tuomo Kähkönen1 and Pekka Abrahamsson2 "Digging into the Fundamentals of xtreme Programming Building the Theoretical Base for Agile Methods" Proceedings of the 29th EUROMICRO Conference "New Waves in System Architecture" (EUROMICRO'03) 1089-6503/03 $17.00 © 2003 IEEE

13) Mauricio Finavaro Aniche, Marco Aurélio Gerosa "Most Common Mistakes in Test-Driven Development Practice: Results from an Online Survey with Developers". Software Testing, Verification, and Validation Workshops (ICSTW), 2010 Third International Conference on, DOI: 10.1109/ICSTW.2010.16, IEEE Conference Publications 2010

14) Roberto Latorre "A successful application of a Test-Driven Development strategy in the industrial environment" 27 September 2013 © Springer Science+Business Media New York 2013, Empir Software Eng (2014) 19:753–773 , DOI 10.1007/s10664-013-9281-9 , Springer (2014).

15) lokman. Rahim, Jon whittle "a survey of Approaches for verifying model transformation" Softw Syst Model (2010) 9:7–20
DOI 10.1007/s10270-013-0358-0 Springer-Verlag 2013