ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Special Issue 15, March 2016

COMPARATIVE STUDY ON TEST CASE PRIORITIZATION

A. Shivaprasanth[1], T. Kirubalakshmi[2], R.Kamalraj[3]
[1 & 2]Student of CSE Department, SNS College of Technology, Coimbatore
[3]AP, Department, SNS College of Technology, Coimbatore

ABSTRACT

Software testing is very important in order to achieve the best software. test cases are necessary for testing.That test case is selected based on the prioritization approach. In this paper different techniques that are available to generate the test cases are listed and a comparative study on prioritization is done. This paper is focused on what is the knowledge that is required by a tester while testing, needs of UML diagram and generated a test case for any real time system.

Keywords – Test case prioritization, Open Dependency, Closed Dependency, categories, history based prioritization,knowledge based prioritization,model based prioritization.

I. INTRODUCTION

Software testing plays a very important role in the software development. Software system is consumed and expensive task [10] [11]. It uses test cases to identify the errors. Test cases can be generated in two ways either manually or automatically. Manual testing involves lots of manual work to test a software, but automated testing consists of testing tools that are used to test the software. The main purpose of software testing is to ensure whether the software works according to the customer requirements.There are different types of the testing methods available and some of them are black box testing, white box testing, unit testing, integration testing etc….depends on the requirement and needs the testing varies .

This paper is about the comparative study on prioritization of test case.In order to find the defects earlier we are going for prioritization process. Test case prioritization is a process of organizing the test case in sequence in order to meet certain goals. The Prioritization is applied in order to achieve the effective test case.There are different techniques and categories available in order to achieve the higher prioritization.This paper tells the different prioritization techniques available, calculating methods and how it varies for different dependency structure and categorized.

II. COMPARATIVESTUDYONPRIORITIZATION

A. TestCasePrioritization

Test case Prioritization is a technique used to find the higher priority test case in order to detect the errors earlier. By assigning a higher priority to a test case we can easily find the errors during the test run. Priority is founded based on graph coverage value. The graph coverage value of a test case is measurement of the complexity dependents of that test case [2]. There are two types of the structures available they are

1. Open dependency structure
2. Closed dependency structure

Test case prioritization varies for these structure.

B. OPENDEPENDENCYSTRUCTURE

1. Total Dependents Of Test Case
2. Longest Path of direct and indirect dependents of test case.

C. CLOSEDDEPENDENCYSTRUCTURE

In closed dependency structure, graph [2]coverage value can be founded in three ways,they are as follows

1. Number of non-executed test cases
2. Ratio of non-executed and executed test cases and find the higher weight and give higher priority
3. Combination of above 1 and 2.

III. CATEGORIES

Test case prioritization [2] is divided into three categories. Based on these categories only best prioritization [2] technique can be obtained. These categories are as follows

1. History based prioritization
2. Knowledge based prioritization
3. Model-based prioritization.

A. HISTORYBASEDPRIORITIZATION

This technique involves collecting the information from the previously used prioritization method. Here are the some of the techniques that are in use, which tells how the best test case prioritization can be achieved [2].

1. Prioritization should have the highest probability of finding errors.
2. Prioritization on the basis of code not tested before
3. Prioritization based on untested code [2]
4. Greedy algorithm

It selects the prioritization based on the highest coverage area.

B. KNOWLEDGEBASEDPRIORITIZATION

This tells what are the human knowledge required to achieve the best test case prioritization. Given below are the some of the knowledge that are required by a tester while searching for a best test case prioritization.

1. Testing Importance of Module (TIM)

TIM is used to find fault proneness. In this technique test cases are split into separate modules [2], after splitting into a module,prioritization is founded based on TIM ranking. So the tester should want to avail of this knowledge.

2. According to Krishnamoorthi and Sahaaya[9] test cases are ranked using 6 factors [2] in order to find the more fault detection. These factors are given below

- customer priority
- requirement change
- implementation
- complexity
- completeness
- traceability

It gives more fault detection then the random prioritization.[2].

3. Tonella tells that best prioritization can be achieved by comparing two –test cases [2].

Example :

4. Srikanth [4] uses four factors to find the best test case prioritization,they are as follows

- Requirement volatility
- Customer priority
- Implementation complexity
- Fault proneness of requirement.

5. Yooet uses[5]the clustering method which reduces the number of comparisations.If there are more than two test cases means it requires more number of comparisations but using this technique the comparisations can be reduced.

Clustering methods involves grouping the set of test cases together.

6. Combinational Interaction Testing (CIT)

It is used by [7] quet in which subsets of test cases are choosen to control the combinational explosion problem.

7. Adaptive Random Testing (ART)

This testing technique [2] is used by Jiang to find the distance measures in which selection can be made based on the random tests. Randomly we are making an selection of an test case in order[2] to calculate the distance.

8. Basanier[8] uses UML models to calculate the prioritization. It consists of the trees structure.

ADVANTAGESANDDISADVANTAGES

APPROACH	ADVANTA	DISADVANTAGE
testing importance of module(TIM)	Easier	Splitting should want to be proper
Krishnamoorth and sahaaya test cases	Fault detection is more	More risk for an tester
Tonella	Easy and simplest way	either leads to success or failure
Srikanth	More efficient	Complexity varies
Yooet	Number of comparisations is less	Grouping want to be carefully
Combinat-ional interactio-n testing(CIT) by quet	Easiest way	success rate is not cent percent.
Adaptive random testing (ART) by jiang	easy understanding	Knowledge of data structure

C. MODELBASEDPRIORITIZATION

Kundu et [2] uses UML diagrams as an model to prioritize. UML-unified modelling language provides the easy understanding and analysis of the system. The UML [9] diagram can be either static or dynamic model. Based on this prioritization of test case can be founded.There are different types of UML diagram and ,they are as follows

1. CLASSDIAGRAM

It consists of class, name, attributes, operations and the relationship between them. We can specify the access specifiers with the help of symbols

# protected
+ public
- Private

2. USECASEDIAGRAM

Use case diagram it will provide the relationship between the actors and use case.It consists of use case, actors, boundary and associating line.

3. SEQUENCEDIAGRAM

It will represents the sequence of operations in the process.It consists of objects,dotted line and straight line.

Dotted lines are used to represent indications and straight line to represent the interactions between the objects.

4. STATECHARTDIAGRAM

State chart diagram consists of sequence of states of objects and the flow of control of one state.In this rectangle represents the states and small circle represents the start state and small white circle with small black circle inside represents the end state.

Start state
End state

5. ACTIVITYDIAGRAM

Activity diagram represents the flow of control of whole system. It depends on the flow of whole process.

6. COMPONENTDIAGRAM

Component diagram explains about the components such as source code, user interface and executable model.

7. DEPLOYMENTDIAGRAM

Deployment diagram tells about the behaviour of the system in runtime.

These are some of the diagrams under UML diagrams.

There are two types of the weight, based on this only prioritization can be calculated. They are

1. Message weight-number of messages between edge nodes
2. Edge weight-number of paths in the sequence diagram that passes through the edges.

Prioritization based on three metrics, they are

1. Sum of message weight
2. Average of edge weight
3. Average of message weight and edge weight for all the paths.

These are the categories that are needed to achieve the best test case prioritization.Weight based and metrics given are used in a graph. It is possible when we generate the graph from an UML diagrams It is the one of the technique for finding best test case prioritization.Below a real time system model illustrated the test case prioritization for an ATM transaction is discussed.

Example-ATMTRANSACTION

The transaction involves the following steps, first user will insert a card then it will be asking for the password if the password is correct transaction will moves to the next step otherwise it will again moves to the previous step and ask us to enter the password again. If password is wrong for more than two times then it will lock a card. In next step it will ask us to select the type of transaction and it asks us to enter the amount if the amount is available in the savings it will perform the transaction or else it will display the error message,that enter the amount that is available in your balance. Then if user entered the amount that is available in our balance it will perform the transaction or else it will quit. Finally user can take the card from ATM machine these are the steps that are available while we performing the ATM transaction.

There are many test cases available in order to perform the testing process and to achieve the efficient software development, the some of the test cases for ATM transaction are as follows while we are inserting a card before it ask us to enter the password there are many inner steps available. It includes verifying the card with bank, whether its original or not, finding the user name, account number before going to a next step. A Test case is to be identified for verifying the card.

In next step it will be asking for a password in that it includes comparing the entered password with password that given to the user. The Password for that card will be stored in a database while comparing it will takes the password from the database then will be compared with entered password. These is the test case for comparing, to check whether the comparison takes place correctly or not.

Tester want to check whether the system performing the operations correctly depending on the password given(right or wrong password).test case for a coding .checking whether code written performing the operations correctly depends on the input given.0

In Next step it will be asking for a type of transaction, test case wants to check whether it moves to correct transaction or not.

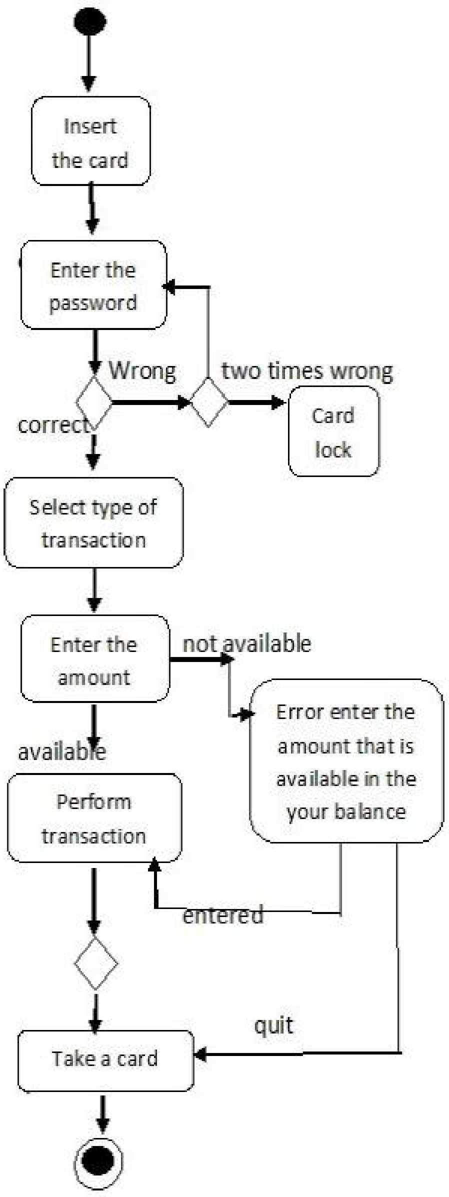
It will asking us to enter the amount if the amount is available in the savings means it will perform the transaction otherwise it will display an error message,enter the amount that is available in your balance.there will be a database for storing the saving amount of the persons,testcase want to check whether retrieval from the database works correctly or not.database will contains all the information we want to check whether the retrieval takes place correctly .test case for checking the retrieval from the database.

Depends on the users command,transaction wants to be performed correctly,after all the operations completes it want to move to the last step that is taking a card.test case wants to check whether the code is written in such a way that it moves to the end operation after completion of all the operations.

Getting a password is very important, so it has highest priority and it is the first test case that needs to executed depends on the prioritization. The Comparing with database having next higher prioritization because it involves checking the password and so it is ranked

TESTCASEPRIORITIZATION

In this first step that is verifying the card, finding the user name, account number is very important so it has highest priority and so is the first test case to executed depends on the prioritization. Comparing with database having next higher prioritization because it involves checking the password and so it is ranked.



ATM Transaction

CONCLUSION

Different approaches given by great research persons does not applicable for all the software. Approaches varies from software to software. So select the approaches correctly based on the software requirement and the complexity of the software .Best test case will leads to the best testing process and best testing process will leads to the best software development, so select the best test case in order to achieve best software.

REFERENCES

[1] Sangeeta Sabharwal, Ritu Sibal and Chayanika Sharma, “Applying Genetic Algorithm for Prioritization of Test Case Scenarios Derived from UML Diagrams”, IJCSI, Vol. 8, Issue 3,
No. 2, May 2011

[2] Shifa-e-Zehra Haidry and Tim MillerUsing “Dependency Structures for Prioritization of Functional Test Suites”, VOL. 39, NO. 2, Feb 2013

[3] R. Krishnamoorthi and S.A. Sahaaya Arul Mary, “Factor Oriented Requirement Coverage Based System Test Case Prioritization of New and Regression Test Cases,” Information and Software Technology, vol. 51, no. 4, pp. 799-808, 2009.

[4] H. Srikanth, L. Williams, and J. Osborne, “System Test Case Prioritization of New and Regression Test Cases,” Proc. Fourth Int’l Symp. Empirical Software Eng., pp. 62-71, 2005.

[5] P. Tonella, P. Avesani, and A. Susi, “Using the Case-Based Ranking Methodology for Test Case Prioritization,” Proc. 22nd IEEE Int’l Conf. Software Maintenance, pp. 123-133, 2006.

[6] S. Yoo, M. Harman, P. Tonella, and A. Susi, “Clustering Test Cases to Achieve Effective and Scalable Prioritisation Incorporating Expert Knowledge,” Proc. 18th Int’l Symp. Software Testing and Analysis, pp. 201-212, 2009.

[7] X. Qu, M.B. Cohen, and K.M. Woolf, “Combinatorial Interaction Regression Testing: A Study of Test Case Generation and Prioritization,” Proc. IEEE Int’l Conf. Software Maintenance, pp. 255-264, 2007.

[8] Sapna P.G. and Hrushikesha “Automated Scenario Generation based on UML Activity diagrams”, International conference on information technology, pp.209-214, IEEE (2008).

[9] Dr. Velur Rajappa, Arun Biradar, Satanik Panda “Efficient software test case generation Using Genetic algorithm based Graph theory” International conference on emerging trends in Engineering and Technology, pp. 298--303, IEEE (2008).

[10] Praveen Ranjan Srivastava and Tai-hoon Kim “Application of Genetic algorithm in software testing”, International Journal of software Engineering and its Applications, vol.3, No.4, pp. 87--96 (2009).