International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 4 Issue: 4 230 - 233

Model Based Prioritization Technique for Component Testing

Ms.Rupali A. More
Department of Computer Engineering
Bharati Vidyapeeth's Deemed University College of Engineering,Pune.
rupalimore2318@gmail.com

Dr.Shashank.D.Joshi
Department of Computer Engineering
Bharati Vidyapeeth's Deemed University College of Engineering,Pune.
sdj@live.in

***Abstract*—** To test modified system through outdated test suite regression test process is required. However, this process of retesting takes lot of time and resources because test suite size has been large. Due to this time, consuming nature this process is not efficient for large test suite. Therefore, there is need of best test case prioritization technique to solve issue regarding retesting for large test suite. This technique has been use to evaluate test cases like higher priority test cases executed before lower priority. Main motive behind this work is to recognize faults at earlier stage so that the debuggers can begin their work earlier. In this paper, we propose a new prioritization technique for test cases to perform regression testing for Component Specific Software System (CBSS). Prim's and Kruskal Algorithm are used.

Keywords-*component; Regression testing, software components, state chart diagram, CBSS, CIG.*

*****

I. INTRODUCTION

Component Specific Software System (CSSS) is an emerging technology that focuses on building systems by integrating existing software components. The component specific software system in which test is been normally conducted on reusable parts. In addition, this CSSS does not require much effort to develop as well as do not require much budget. Therefore, to design qualitative component specific software efficient testing is important. There are many test categories available like test applied to acquire selective portions of code rapidly. In component specific software system the components are executed in different circumstances like those that by using programming languages, implemented by operational platforms distributed on any place, components might be developed in house or by any third party. As the large size of a test suite has been use for system retesting, it tends to consume a large amount of time and computing resources; it may be for hours, or even days. So one of the issue that developer faces during the retesting of the system is ordering test case for execution. Test case prioritization tries to address this issue. Testing of software units is major issue in component specific software system. When we want to modify or add a component and apply the regression testing, it incurs more cost and time. So to reduce these two factors we can use a test prioritization technique which is based on two criteria like maximum state changes and maximum data base access occurred by a test case during component interaction scenario. Test case prioritization orders test cases for execution so that the test cases with the higher priority, based on some criterion, are executed before lower priority test cases. In this project, our try is to apply prioritization of test to retest component specific software. This system will use reusable units as base to design complex framework of software, which provide quality and productivity. This base approach is widely used for large software framework. This system has a set of some loosely coupled units, which allow plug and play. In addition, CIG approach has been use to generate test cases automatically. Reminder of this paper articulates details in section II it review some previous work, section III clear idea of project, section IV shows results and section V will conclude our discussion.

II. RELATED WORK

In existing System, code-based test case prioritization is been done. In code base test case prioritization, source code of the system is used to prioritize the test cases. However, this process is time consuming and it will not detect early fault as compared with proposed system. A CIG has been constructed from a state chart diagram and new test cases are generated to test the component composition [1]. Prachi Batwara et al. in 2010 [2], proposed combination of greedy algorithm and genetic algorithm to prioritize test cases and compared performance. But in this greedy algorithm not able to generate test cases for each requirement. In next search to prioritize test cases, only genetic algorithm was used by ritu sibal et al. to recognize major path clusters [3]. In next research [4] by Bindhyachal Kumar Singh genetic algorithm used to generate test cases automatically and to satisfy path coverage. To test path test cases generated through genetic algorithm. Similarly, ruchika malhotra et al. [5] also used genetic algorithm for test cases by using test case history. In this case if test case gone through more coding line then fitness function generate high value. According to that, if test case has high value will get first priority. Therefore, this approach mostly gives effective results. In model-based test case prioritization, a system's model has been use to prioritize the test cases [6]. There is model used to observe nature of system. Model based

test case prioritization help to enhance fault detection at early stage compared to the code-based test case prioritization. Model-based test prioritization may be an inexpensive alternative to the existing code-based test prioritization methods. This method is somehow not efficient to correct/incorrect data given by the testers/developers. In 2013 research, proposed APFD metric for test case prioritization for code coverage [7]. In same year, Haiqiang Li et al. [8] proposed a new model Contract-Based Combined Component Interaction Graph Generation, which will be effective for test cases creation of the state transition among components and data communication. Shah nazir et al.[9] proposed a mechanism to prioritize test case based on fuzzy network that is (FANP) for component testing. This method is combination of fuzzy and analytic approach. Paper surveys various previous techniques for test prioritization.

III. PROPOSED WORK

In this project work, we are designed system for test cases prioritization. Our testing criterion has been base on components. According to proposed, system uses model based test case prioritization testing. Here our motive is to reduce time of retest. Therefore, by applying logic high priority test cases served first compare to lower priority. The model based test case prioritization improves the early fault detection as compared to the code-based test case prioritization. We are using following algorithm for prioritization.

Prioritization function:

Input: TS=Test Suite ie T1, T2…Tj
C= of components. i.e. C1, C2, C3…Cm
S= set of states. S1, S2…Sn
t=time.

Result: Test Prioritized T'

Start
Set T' as empty
For each Tj belongs to Ts
do
all Countinter, Countintra, DBtotal, DBdirect DBindirect initialize to zero.
for(k=1; k<m; k++) // m: total components
for(i=1; i<n; i++) // n: total states
if(Ck(Si,Transi,si+1)) $\longrightarrow$ Ck+1(Si)
//one component(Ck) state activates the Ck+1 .
Countinter ++;
// inter component incremented by one
else
Countintra ++;
// intra incremented by one
If (DBaccess = 'TRUE' && DBaccess _Type = = 'DIRECT')
DBdirect = DBdirect ++;
else
DBindirect = DBdirect ++;
DBtotal:= DBdirect + (2* DBindirect)
Obji =[Countintra +2*(Countinter) + 3*(DBtotal)] /t ;
end all for loops
T sort in descending order based on values of Obji
Set T' =T
Result T'

This used algorithm has been found to be very effective in; maximizing the objective function and Minimizing the cost of system retesting.

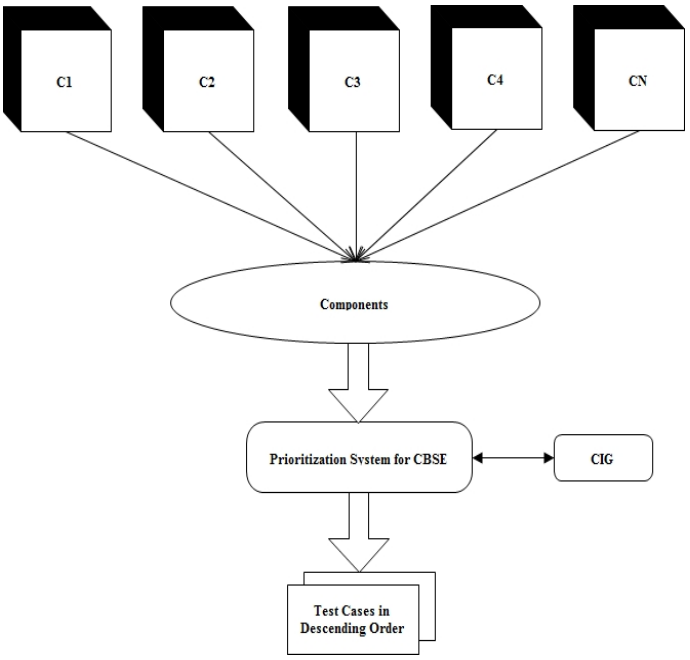


Figure: Test Prioritization System Architecture

IV. RESULT ANALYSIS

This system is for test prioritization, this section give overview of system. For this project, we used library management system and student management system to apply test prioritization.

1. Home page

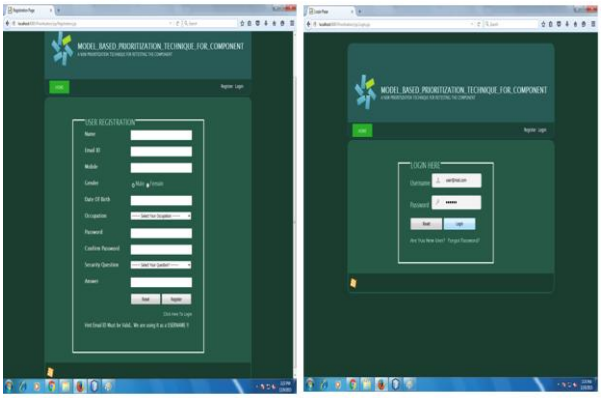
2. Registration and login

In this module user can make registration and login using their id and password. Only registered user can

 ISSN: 2321-8169
230 - 233

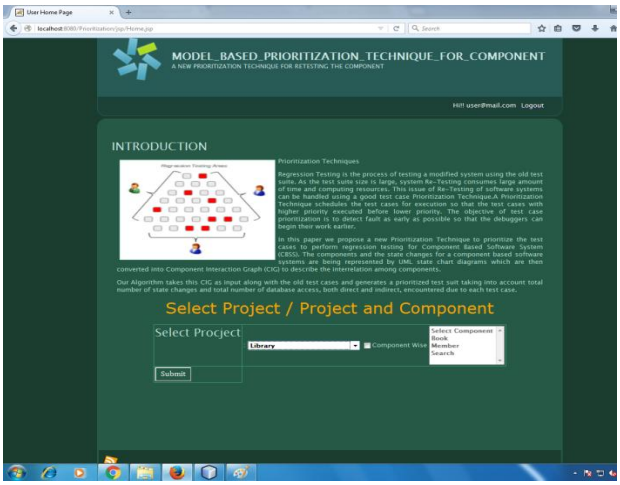
check and get the prioritization test case result of a system.

Register Login



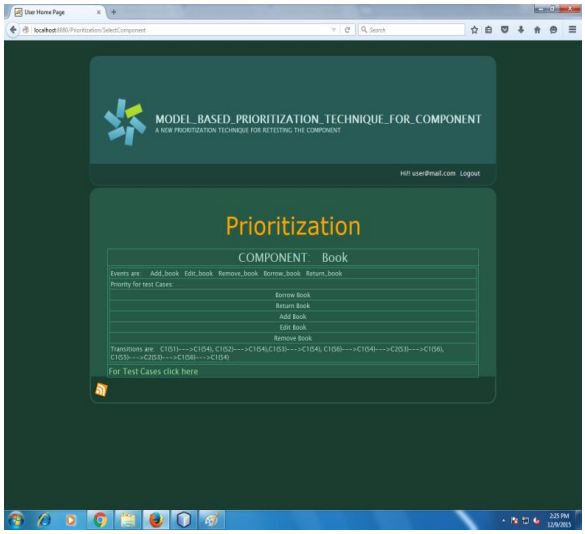
3. Component selection

In this window, we can select from two projects and component criterion is given as book, member and search. Student management system having student, view and account are components.

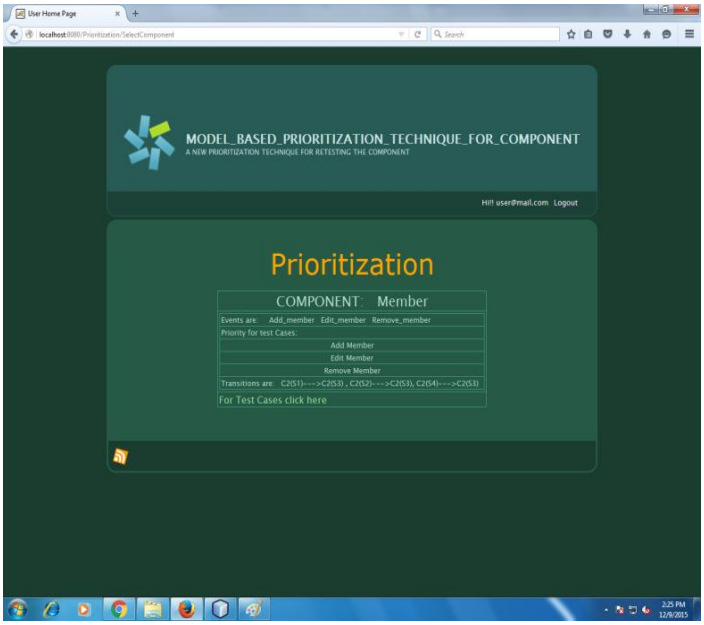


4. Component view

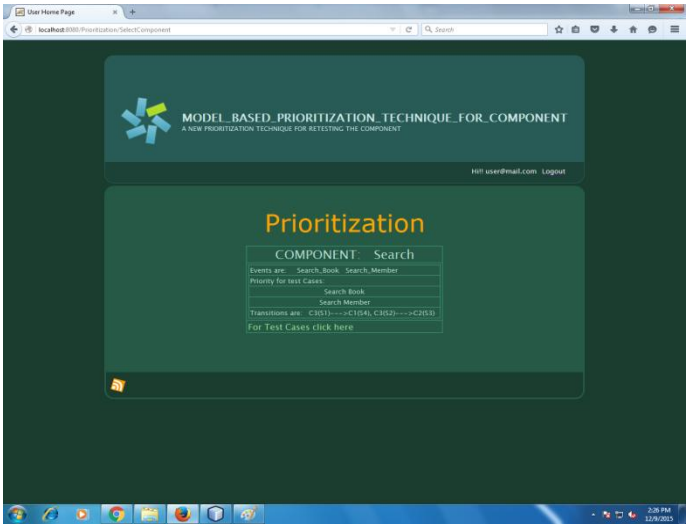
A. Book



B. Member

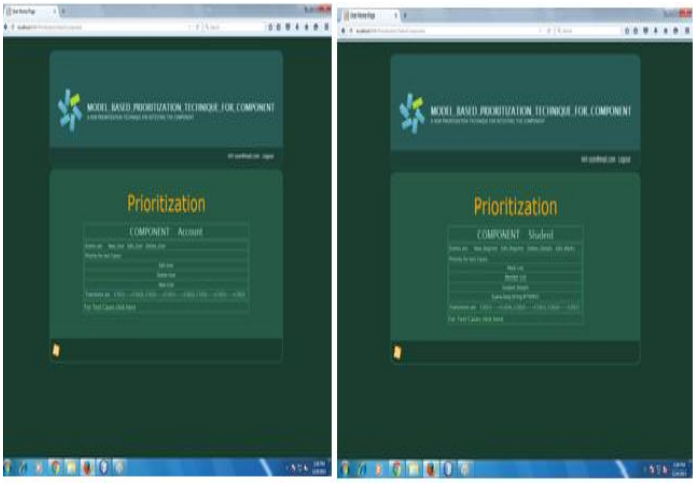


C. Search

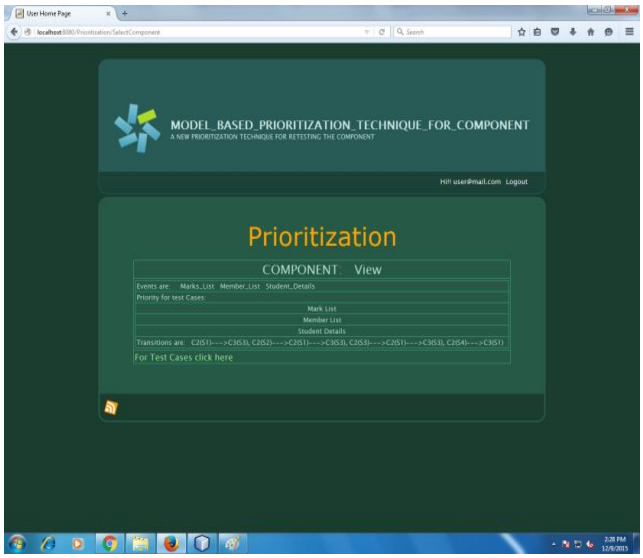


5. All components of student management system

A. account and Student



B. View



In this view of results, we used two existing projects, if we select particular component according to that we will get different test cases so we can perform prioritized test case. In addition, there is provision to select all component test cases instead of individual. Higher priority test case served first. Our output will be prioritized list of test cases.

V. CONCLUSION

This project has been intended for test case prioritization, which overcome pitfall of previous testing techniques, and gives best results in less time. To achieve this goal we reviewed some previous techniques and by using idea of prim's and kruskal algorithm, we articulated our system. Now system is specific to components in future we can design system based on events driven software framework.

.

REFERENCES

[1] Arup Abhinna Acharya, Sisir Kumar Jena," Component Interaction Graph: A new approach to test component composition", journal of computer science and engineering, volume 1, issue 1, may 2010.

[2] Prachi Batwara , Yashika Sharma, "performance comparison of test case prioritization search algorithms" 2nd National Conference in Intelligent Computing & Communication Organized by Dept. of IT, GCET Greater Noida, india, 2010.

[3] Ritu Sibal , "Applying Genetic Algorithm for Prioritization of Test Case Scenarios Derived from UML Diagrams" IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 3, No. 2, May 2011 ISSN (Online), 2011.

[4] Bindhyachal Kumar Singh, "Automatic efficient test data generation based on genetic algorithm for path testing". International Journal of Research in Engineering & Applied Sciences, Volume 2, Issue 2 February 2012.

[5] ruchika malhotra, abhishek bharadwaj," Test case priortization using genetic algorithm", International Journal of Computer Science and Informatics ISSN (PRINT): Volume-2, Issue-3, 2012.

[6] Prateeva Mahali, Arup Abhinna Acharya, " Model Based Test Case Prioritization Using Uml Activity Diagram And Evolutionary Algorithm", International Journal of Computer Science and Informatics, ISSN (PRINT): 2231 – 5292, Volume- 3, Issue- 2, 2013, pp. 42-47

[7] K. VimalaDevi, "Effectiveness of Test Case Prioritization using APFD Metric" International Conference on Research Trends in Computer Technologies (ICRTCT - 2013) Proceedings published in International Journal of Computer Applications® (IJCA) 2013.

[8] Haiqiang Li, Min Cao and Jian Cao, "Contract-Based Combined Component Interaction Graph Generation", Springer Science+Business Media New York 2013, pp. 781-789.

[9] Shah Nazir, Sara Shahzad, Neelam Mukhtar, Humaira Khan, Islam Zada, Muhammad Nazir and Rohul Amin, " Test case prioritization for components using FANP", Life Science Journal 2014