AUTOMATED TESTING IN DEVELOPMENT PHASE

SUNIL L. BANGARE1, SACHIN M. KAMBLE2, PALLAVI S. BANGARE3, ABHIJIT V. NAIK4

1, 2, 3, 4 Department of Information Technology, STES’s Sinhgad Academy of Engineering, Pune-48, Maharashtra, India.

sunil.bangare@gmail.com

Abstract:

In software development the applications are tested in testing phase of software development process. So testing of application is not possible without complete development of module/application. It takes additional time in completion of software development. So this paper proposed the model for tool which provides the way to developer to test his code/application in development phase itself. The model also helps in java API (application programmable interface) testing. With this tool, developer can able to test his code/module automatically considering all the aspect of testing. In this approach predefined test cases are loaded for testing, and thousands of test cases are run at same time and application is tested by developer. So it helps in regression testing. Hence it helps in reducing software development period. Ultimately it saves the people resources, hardware and software resources.

Keywords: Automation, API testing, test case, regression testing, custom test case, report.

1. Introduction

The software development process consists of different phases i.e. Requirement, analysis, planning, design, implementation, testing, deployment & maintenance. So the software organization has main focus on implementation and testing to satisfy the actual working of software. In current trend the development and testing are completely different phases. So in software development process we have to switch next phase after completing previous phase. This scenario is strictly applicable to implementation and testing phase. The proposed model is demonstrating the strategy in which both phases i.e. implementation and testing are to be carried out at same phase considering testing as automated testing. The tool based on the proposed model will help to implement same strategy. So the testing of software can be carried out at same phase i.e. development phase. This will reduce the 80% of testing efforts. By which all testing aspects will taken into consideration. And tool based on the same strategy will help in the regression testing.

2. API (Application Programmable Interface) Testing

The basic process of testing an API is to create the test harness which calls functions in API with interesting data [3].

Automation in API testing helps in running and accessing many test cases. Several aspect of API testing can be automated:

- Defining test cases.
- Data selection.
- Execution of large test cases.
- Result checking. [2]
As shown in fig, once the developer implements the module the task is to assign the test cases to module and test it by the system. Here we can assign the thousands of test cases to particular module and will check that test cases for that module. After testing those test cases showing error message or say bug are considered for editing by the developer. And retesting is considered for all test cases again. This model helps in regression testing. The software development process is a difficult and modularization can makes it more complicated. This is the challenge to measure the quality of objects oriented software modularization. Modularization of object oriented code is distribution of the software in to modules and these modules should communicate with each other through some application programming interface (API) [5]. The main problem is of communication between the modules. Generally this should be done through some application programming interface (API) [6].

3. Regression Testing

Regression testing is the executions of some have not propagated unintended side effects [1]. The tool based on the proposed model is helpful in regression testing where many numbers of test cases can be executed shortly and according results are observed. So this approach reduces the headache of user to execute each test case separately and side effect on functionality. With this approach all testing aspect is inspected by the developer so there is a lack of chances of being bug in the software. So each module of application is implemented and verified in development phase only. Mainly this approach of testing helps in controlling the regression testing.

4. UML View for Proposed Model

The activity diagram shown below focuses on the test cases on the flow of system while assigning the test cases to the application code where it is clearly indicating that we can load test cases which already generated and we can define the test cases. After an testing if result is satisfactory i.e. if actual result and expected result is match completely then the system is going to pass test cases otherwise bug report is generated and further test cases implemented sequentially.

---

Fig. 1: Automated testing in development.

Fig. 2: Activity diagram for the proposed model
5. GUI of tool based on this approach

The model is mainly defined for development tool so it is provided with an code editor so developer can implement the code and test it. In testing part, the functionality is to custom the test cases assign and test the application or module automatically. As shown in the above snap shot the developer is provided with a facility to load previously generated test cases for current application so user is not supposed to define new test cases if test cases are present already. Hence for the testing purpose the reusability is possible with this approach.

6. Sample output generated

As shown in above diagram every test case having its id and all test cases are applied on the code and hence all the test cases are tested by the system according result are compared and report is generated. In case of fail the test cases editing is done and all the test cases are executed again. Hence automatically regression testing is done [8].

7. Flow of the Proposed System

![Flow diagram automated testing approach in development.](image-url)
This is the flow of the module. User can create test cases with the help of classes, methods and input/output. The test cases are assigning to the module. As shown in the diagram compiler interface are used in editing the code and compile it after that by referring the class file generated by the compiler is given to testing module where the testing method is executed. With the help of reflection, computer program can modify and observe its own structure at runtime. This module can test any number of java files with the help of automated testing. This module is also featured by compiler interface. Test cases are tested in automated fashion and finally test report is generated which helps user to guide about the anomalies in early stages. All Intermodule function calls are routed through the published API [7].

7. Advantages
- Task Reduced: This module helps to reduce the task of testing the software as majority of testing is done at the time of the development.
- Faster: Automated Testing run tests significantly faster than human beings.
- Comprehensive: This module can build a suite of tests that covers every feature those are required in application.

8. Applications
- Software organization: In the software organization for the development of the software in development phase developer can use the tool based on the model proposed. For editing and testing in development phase.
- Testing phase: In software organization tester can use the tool based on proposed theory for testing an application. Again the quality assurance team can use the same tool to verify the code.
- Online coding competitions: This module can help users to add their own test cases. This module can help normal user to add new challenges to the coding competitions.
- Academics: This module can help students to test the test cases while developing them. So students can correct the anomalies in their test cases in early stages.

8. Conclusions
The proposed model for an Automated Testing of application will help in software organization to reduce efforts in repeated task. Hence proposed model helps in Rapid Application Development. Proposed model also helps in reducing testing work in application development. It is possible to API testing with the tool based on proposed theory. Reusability also becomes possible n test cases we can use the old test cases for similar type of application.

References