ISSN XXXX XXXX © 2019 IJESC

Research Article Volume 9 Issue No.3

Automated API Testing

Isha[1], Abhinav Sharma[2], M. Revathi[3]
Department of Computer Science and Engineering
SRM Institute of Science and Technology, Chennai, India

Abstract:
Application programming interface, or API, is a piece of code that enables two software components to interact. Recent software applications are becoming distributed across various servers that interact with back-end utilities through standardized interfaces. Software applications provide a part of their data and business logic through APIs with the end goal to produce extra revenue called API economy. The number of APIs are expanding exponentially with every year. Thus, directing automated tests on the APIs has turned into a pivotal step in software development process since malfunctioning or ineffective APIs can result in lower acquisition of product and ultimately loss in revenue. API testing is crucial to ensure functionality, reliability, security and delivery of business logic. Major challenges faced during automated API testing include sequencing API calls, comparing unpredictable response, handling planned failed test cases, parallel API execution and handling parameter reliance. This paper presents proposed automated API testing tool and literature survey of automated API testing.

Keywords: API, Web API, REST API, automated testing, automated API testing, JSON

I. INTRODUCTION

[1]API testing is crucial to ensure functionality, reliability, security and delivery of business logic. [2]Manual API testing is a very time consuming and tiring process. Since manual API testing is done by human, there is high risk of errors and mistakes being made. It is impossible to manually compare large amount of data. Manual API testing is not suitable for large organizations and time-bounded projects. Even one malfunctioning API can affect the whole product. Therefore, manual API testing is deprecated and automated API testing is preferred. Automated API testing is not only a fast process but also reduces chances of errors. Automated API testing does not require physical presence of tester and can be performed anywhere and anytime. [3]Web APIs are further classified as SOAP (Simple Object Access Protocol), RPC (Remote Procedure Call) and REST (Representational State Transfer) depending upon the data exchange format. REST or RESTful APIs are most widely employed APIs and hence, proposed API testing tool exclusively targets RESTful APIs. RESTful APIs are based on REST technology, a paradigm that defines set of protocols for communication used in web services development. [4]A RESTful API uses HTTP methods such as GET, POST, DELETE and PUT. GET API is used to retrieve a resource. PUT API is used to change a resource. POST API is used to generate a resource. DELETE API is used to remove a resource. [5]RESTful API response consists of : program, version, release, datetime, timestamp, HTTP status code, message, data. JSON (JavaScript Object Notation) format is used for delivering RESTful API responses. Most of the existing testing software use HTTP status code to test a collection. This method is not useful as tester does not get the exact response comparison which is of utmost importance in testing. The proposed automated API testing tool validates an API based on comparison between expected JSON response given by tester and actual JSON response received from the server. This technique eliminates any threats involved in an API response and gives more insight about an API to the tester. In certain software services, API calls should be executed in a specific sequence. Parallel API execution disorders the API sequencing calls and results in erroneous testing. To prevent such disorderness, proposed API testing tool identifies sequenced API calls and executes them in given order and not in parallel. Proposed automated API testing tool identifies all independent API calls and executes them in parallel. The proposed automated API testing tool employs banker's algorithm to prevent any deadlock in the sequenced API calls.

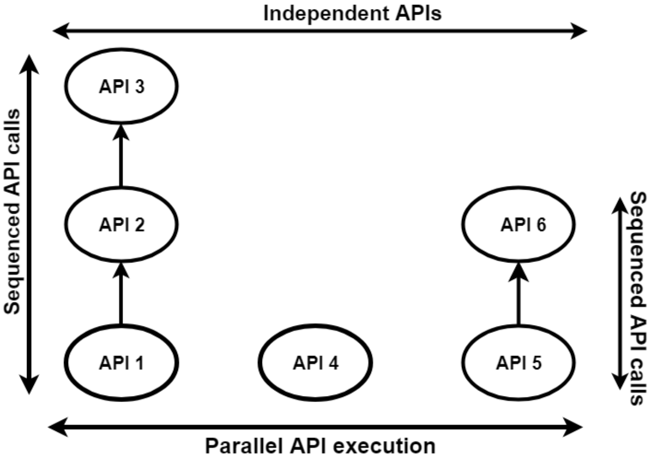

Figure.1. Orderly execution of sequenced API calls and parallel execution of independent API

As depicted in the fig. 1., the proposed API testing tool identifies all the sequenced API calls and executes them in order and executes independent APIs parallely to maintain orderliness. In many cases, the input parameters to an API call depends on the response obtained from some other API call. This arises two problems - sequenced API calls and extracting input from response. Existing API testing tools does not provide support for parameter reliance and hence does not provide 100% automation. The proposed automated API testing tool parses the JSON response and extracts the required parameter value for next API call. This process increases the automation in API testing. The tester also tests the APIs for wrong inputs to check if API handles it gracefully. In some cases, pass and fail test case may return same HTTP status code. Hence, this intended failed test case testing is not possible exclusively based on HTTP status codes but requires to analyze the API response. In many cases, the tester might not know the exact response of an API. Like in cases where an API response contains a uniquely generated number or string. In such cases, proposed automated API testing tool provides tester an option to completely ignore the field, check if the field exists in the response while ignoring the value of the field or parse the field value with a regular expression. The tool also deploys the testing logic in the cloud and thus enables testers to schedule API test anytime. The proposed automated API testing tool will conduct the test on the given API test suite and produce test results including all passed and failed APIs. Based on the severity of number of malfunctioning APIs, the proposed system will alert the tester about the risk via email and message.

II. LITERATURE SURVEY

This section analyzes various research paper published in the field of automated API testing. [6]The paper discusses importance and impact of automated API testing in the product development phase. In Software Organizations, automated API testing is used to do the manual testing automatically at the time of development. Developer can add his own test cases and he can test the module. Accordingly reports will be generated if API contains any error. The appropriate error message will be displayed in the report. This is useful because developer can change his API according to report generated by the system. We can reduce maximum testing work of tester at the time of development itself. This generic API testing tool will allow users to define test cases, manage test cases, perform individual testing, perform bulk testing and manage test reports. [7]The paper discusses the impact of API technology. By using API technology, we can make a very strong back end, and it can be scalable and the data can be used to render on a mobile device or even a smart TV. Thus, helping to separate the front end environment and backend environment completely. [8]This paper discusses scope of automated API testing. Moving from manual testing to automation framework for API testing, increased lot of scope in testing. The journey from manual to automation and stabilization of code gave good results in managing the web services.

- Opportunity for every QA to learn and enhance programming knowledge.
- With the stable framework, an application can be more stabilized and can deliver good product to customers.
- API testing also gives the deeper insight into the product which in turn helped QA to increase the product knowledge.
- Analyzing the failed test cases helped in detecting defects at the earlier stage. Early detection of defects reduced rate of defect detection at the GUI level.

[9]The paper discusses various types of test cases to be created for automated API testing.

- Input validation - Test the API with different input parameters and verify the response. The data, response code, message etc should be correct for each set of input parameters. Does the API return correct HTTP error codes like 200 for valid and 400 for invalid input parameters?
- JSON format validation - Verify that the JSON or XML response of the API is correctly structured.
- Business Logic - Is the API doing the task it is supposed to do? Suppose we have a fetch Balance API that gives the current balance for a user. If it returns the correct HTTP response code, but the actual balance being returned for user is of previous month, the business logic of this API is broken.
- Negative test cases-Hit the API with incorrect/invalid parameters, missing/extra values, null values for mandatory fields and observe the output. Is the API able to gracefully handle unreasonably large amount of payload data, special and non-ASCII characters, long strings and integers, incorrect data types for parameters? Are there any buffer overflow conditions? How does it behave in case of conditions like timeouts, server failures, etc? Is the exception handling mechanism in place? Are the error messages clear and relevant?
- Reliability tests - Check whether the API can consistently return correct response, or do response failures occur often.
- Call sequencing checks - If the output of the API being tested includes modification of a data structure, change of a resource state (Like a DB record), firing of an event or call to other APIs, this should be functioning correctly.
- Security testing - Every business critical API should undergo security testing techniques, to make sure that its code and services cannot be accessed and utilized by unapproved clients.

III. PROPOSED SYSTEM

A. *Description*

The proposed automated API testing tool is an-easy to use tool that helps to automate the process of RESTful API testing.

B. *Existing System*

Below mentioned are the major limitations of the existing system.

Limitations

- No support for code-less testing.
- No support for parallel execution of APIs while maintaining the orderliness of API calls.
- No support for handling unpredictable JSON response.
- No support for sequencing API calls.
- No support for pass/fail test case judgement.
- No support for scheduled tests.

C. *Proposed System*

The proposed automated API testing tool is an-easy to use tool that helps to automate the process of RESTful web API testing. The proposed API testing tool provides a simple to use GUI based testing tool that enables tester to create an API test suite. As depicted in fig. 2, each test suite contains modules and each module comprises of test cases where every test case has a set of APIs with expected JSON response for each API. The tool executes the APIs in the servers and compares the actual JSON response with expected JSON response. The tool handles sequenced API calls, executes independent API calls parallely, check for deadlocks in API calls, handles unpredictable responses and parameter dependency. The tool provides crisp report of the test with all test-cases failed or passed. The tool also provides feature to schedule the testing, that will be performed automatically in the cloud without the presence of tester. The tool generates test report and sends it to the tester. The tool also alerts the tester in case of critical API failures. The tool doesn’t require tester to know any programming language. The tester should have basic knowledge of REST API and its working.

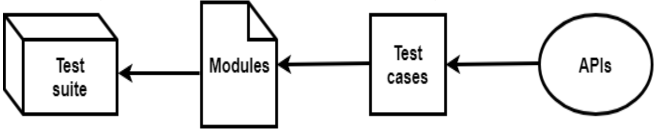

Figure.2. Hierarchy of test suite.

Advantages

- Code-less API Testing.
- Sequencing API calls.
- Unpredictable JSON response handling.
- Parameter dependency.
- Scheduled tests.
- Parallel API execution preserving orderliness.
- Generic testing tool.

Table.1. Comparison between Existing and Proposed automated API testing

Features	**Existing System**	**Proposed System**
Code-less API Testing	No	Yes
Sequenced API calls	No	Yes
Unpredictable JSON response handling	No	Yes
Parameter dependency	No	Yes
Scheduled tests	No	Yes
Automatic testing	Yes (single API)	Yes

IV. WORKING SYSTEM

The proposed automated API testing tool provides a simple to use GUI for creating test suite. The tester can create custom test cases without any programming language. The tool will accept APIs and expected JSON response from tester. The tool will compare the actual JSON response and expected JSON response to validate an API. The business logic of the testing tool is written in java programming language exploiting Apache HttpClient library for http connections. Electron framework is used for building the desktop tool in order to make the tool platform independent. The tool uses Jackson's built-in tree model feature to parse unknown JSON response in order to make the tool generic. After parsing both expected and actual JSON response, the tool performs comparison. In case of unpredictable JSON response, the tester can give command to ignore a field, check if the field is present in the response while ignoring the field value or match the field value with a regular expression. The tool generates a crisp report in .doc and .pdf extension with all passed and failed test cases with reasons of failure (response mismatch). This enables tester to gain more insight in the API and improve API responses making the API more effective and efficient. The tool exploits multithreading in java for parallel API execution. The tool deploys the business logic in Amazon AWS to allow tester to schedule tests. The tool generates report of the test. Based upon the severity of number of API failures, the tool alerts the tester via email and message.

V. APPLICATION

The number of APIs are increasing exponentially with every coming year. [10]The Programmable Web directory passed the 19,000-API mark in January of 2018.

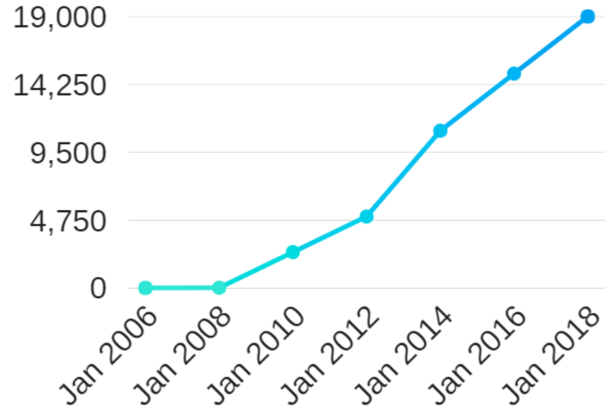

Figure.3. Programmable Web API directory

According to red hat shares, 88% of enterprises use APIs. Hence, the proposed automated API testing tool can be used in an enterprise that employs APIs - public or private. The tool can be used ensure functionality, reliability, security and delivery of business logic.

VI. RESULT

The proposed automated API testing tool solves all the major challenges in automated API testing like parallel execution of APIs, handling unpredictable JSON response, sequencing API calls, pass/fail test-case judgement and comparing response. The tool also helps tester to schedule the tests. Hence, the proposed automated API testing tool reduced the time required by 90%, people required by 95% and cost by 70% as compared to manual API testing. The tool also achieved 95% automation in API testing.

VII. CONCLUSION

The proposed automated API testing tool is a fully automated RESTful API testing tool that solves all the major challenges of API testing automation and provides a simple to use GUI with support for codeless testing, parallel execution of APIs, handling unpredictable JSON response, sequencing API calls, pass/fail test-case judgement, comparing response and scheduled test. The tool eliminates efforts of API testing thus enables tester to gain insight in the product's APIs and responses. The tool helps to ensure the functionality, reliability, security and delivery of business logic. The proposed automated API testing tool helps make product more stabilized and deliver good product to customer. Efficient APIs results in higher acquisition of product and ultimately increase in revenue.

VIII.REFERENCES

[1]. https://www.linkedin.com/pulse/importance-api-testing-ankur-gupta/

[2]. https://www.gcreddy.com/2014/01/disadvantages-of-man ual -testing.html

[3].https://www.mulesoft.com/resources/api/types-of-apis

[4]. https://www.restapitutorial.com/lessons/httpmethods.html

[5]. https://restful.io/an-introduction-to-api-s-cee90581ca1b

[6]. Mithilesh Tarkar and Ameya Parkar. APIs and Restful APIs. International journal of Trend in Scientific Research and Development (IJTSRD),

[7]. Sunil L. Bangare, Seema Borse, Pallavi S. AUTOMATED API TESTING APPROACH. International Journal of Engineering Science and Technology (IJEST). BANGARE, SHITAL NANDEDKAR.

[8]. Asha KR and Shwetha DJ. API Testing: Picking the Right Strategy.

[9].https://www.linkedin.com/pulse/rest-api-testing-from-manual- approach-automation-saurabh-agarwal/

[10]. https://www.programmableweb.com/news/research-shows-interest-providing-apis-still-high/research/2018/02/23