

Non-Functional Characteristics and NFT Testing of Container Applications

Shadab Alam Siddiqui
Research Scholar:
Department of Computer Science
Aligarh Muslim University
Aligarh, Uttar Pradesh, India

Dr. Tamanna Siddiqui
Associate Professor:
Department of Computer Science
Aligarh Muslim University
Aligarh, Uttar Pradesh, India

Abstract—Container applications are complex in nature as they are a collection of micro-services running in sync to achieve the application's desired functionality. This application build structure makes the Non-Functional characteristics of the container applications more relevant and a focus area of testing for the application. In this paper, the authors have tried to investigate all the possible non-functional areas of container-based application. The paper further delves into the nuances of non-functional testing in general which would become an integral part of the Non-Functional Testing (NFT) of container-based applications.

Keywords— Containers, Non-Functional Characteristics, Testing, Non-Functional Testing (NFT).

I. INTRODUCTION

Application development has now been transitioning to container-based architecture from Virtual Machines. This is because of several advantages that a container brings in comparison to a virtual machine [1]. This transition is taking at a fast pace and the architectural change is phenomenal, which brings to the fore the overall applications behavior apart from just the functional part. The functional part of any applications remains constant on any architecture be it traditional hardware, cloud, virtualization or containers. What changes with change in architecture is the surrounding environment, infrastructure etc. and the characteristic these changes makes an impact on, is termed and categorized as Non-functional aspect of application. This change makes it imperative on part of the software or application developers to have extra focus on Non-Functional testing of the container-based applications.

Functional testing of any application refers to the feature set of the application that it intends to provide to the end user of the application. Irrespective of the platform, environment or other external variables, these feature sets or the functional part of the application remains constant. To illustrate it with an example, a railway reservation application's primary functionality of booking tickets remains constant irrespective of whether the application is mobile, cloud or traditional. This stands true for all applications and thus the intended functionality is always rigorously tested and it is a continuous process where even after the application is in production we see number of new releases which could include patches to fix existing bugs application supporting new functionality.

Non-Functional testing of any application encapsulates several external and behavioral parameters and traits of an application. Contrary to Functional behavior, non-functional

behavior of an application fluctuates and changes depending upon the platform, environment and other external variables around which the application is built upon. Taking similar example of the railway reservation application, the non-functional behavior is bound to change based on the surrounding eco-system it is built upon.

II. RELATED WORK

A. Containers

Containers are referred to as packaging units of applications that are built using the concepts of micro-services. Container can also be related to virtualization which extends the concept of virtualization to OS level whereas normal virtualization aims at virtualizing hardware resources. Several characteristic and definition of container along with difference between containers and VM are elaborately define in [1].

B. Application Testing

Application testing is an integral part of any software or application. Without proper and complete application testing it cannot be made live or deployed for production. Broadly application testing can be categorized in two different branches of Functional testing, where the functionality of the application is tested thoroughly and the other being non-functional testing, where the application is put under test because of the surrounding environment and its behavior with respect to changes in these characteristics.

C. Non-Functional Testing

Non-Functional Testing (NFT) is a collective term given to test various non-function traits and parameters of any application. In [3], the author of the article has very effectively and in detail, defined various non-functional attributes, methodology to test these attributes comparison between functional and non-functional testing and the importance of Non-functional testing.

D. Why Containers and its Applications Non-Functional Testing

Adaptation of Container based application has been growing at an aggressive speed because of its benefits over cloud and traditional applications. Containers development methodology is based on microservices architecture which is complex but adaptable and more manageable when compared to the traditional monolithic applications.



Fig. 1. Container revenue and projected trends

Joe [4] has done an extensive research about the growth of containers over the years since 2015 and have predicted its growth path till 2020. He estimates the container market to grow around 40% on a yearly basis.

Non-Functional testing of container-based applications is to extend the NFT to container-based applications. Container based applications being a fairly new and a fast-growing concept which aimed at better Life cycle management and performance of application without utilizing much of resources. Thus, important characteristics needs to be tested thoroughly for such applications to be successful.

III. PROPOSED WORK

In this work, the authors intend to propose the non-functional attributes and characteristics of the Container based applications that should be an integral part of any such applications. Each non-functional characteristic's relevance will be discussed and presented to bring forth its relevance in the overall testing of the application.

A. Non-Functional Characteristics

Below are the proposed Non-Functional Characteristics that should be mandatorily tested for all container applications –

1) Scalability and Performance –

Scalability testing refers to the expansion (scale-up) and contraction (scale-down) of the application as per the real time traffic. This ensures that the application uses optimum resource and thus helps the provider to use its hardware resources more efficiently without effecting the applications run time.

Another by-product or extension of scalability testing is performance testing. While testing applications for scalability, it is deemed that the applications performance also be tested with respect to the following:

- End user experience of the application under stress or load.
- Limit of application vis-à-vis resources allocated or in short capacity of the application.

David [5] suggests 3 best practices to test and evaluate performance of any container-based application:

- Service Oriented approach
- Testing services independently and
- Right size container performance testing.

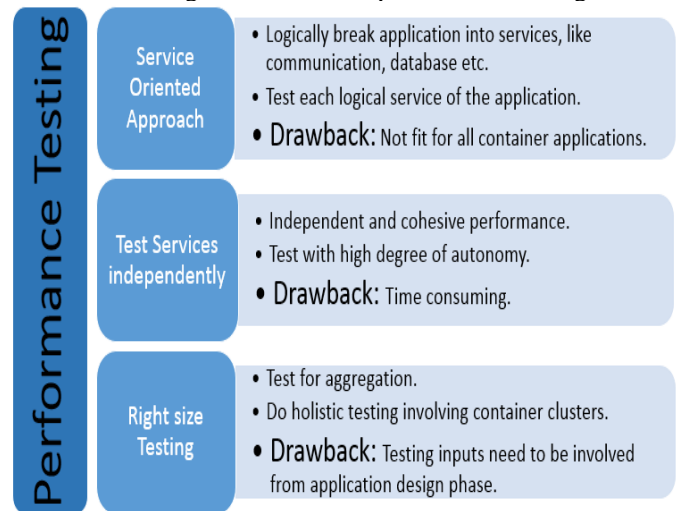


Fig. 2. Performance testing approaches

2) Stability and Robustness –

The criteria set forward to measure the stability of an application is the length or time duration of the application up-time when the application is put through stress, load and different environments.

Robustness testing is an extension of stability where the criteria measured for robustness is to test the application's strength under different load and stress scenarios. This measures that the application is not behaving abruptly when put under duress and responses of the application are as per expectation.

3) High Availability and Geo-Redundancy -

High availability (HA) refers to the application's availability irrespective of system, hardware or any other resource or site failure. It makes sure that the application is redundant in such a way that failure of application instances because of any external reasons does not impact the availability of the application. In simplistic terms HA can also be defined as a property where the application does not have a single point of failure. Kanso et. al [2] in their article have in detail examined the various HA mechanisms with limitations.

Geo-Redundancy of an application refers to the application's availability when an entire data center or a cloud environment where the application is hosted goes down because of any reasons like power outage, natural calamity etc.

The below diagram presents a basic view of the HA and Geo-Redundant architecture of a Container based application environment. The diagram is simplistic in nature as it hides the complexity of several components involved in container and its orchestrator. It also abstracts the integration flow between the components.

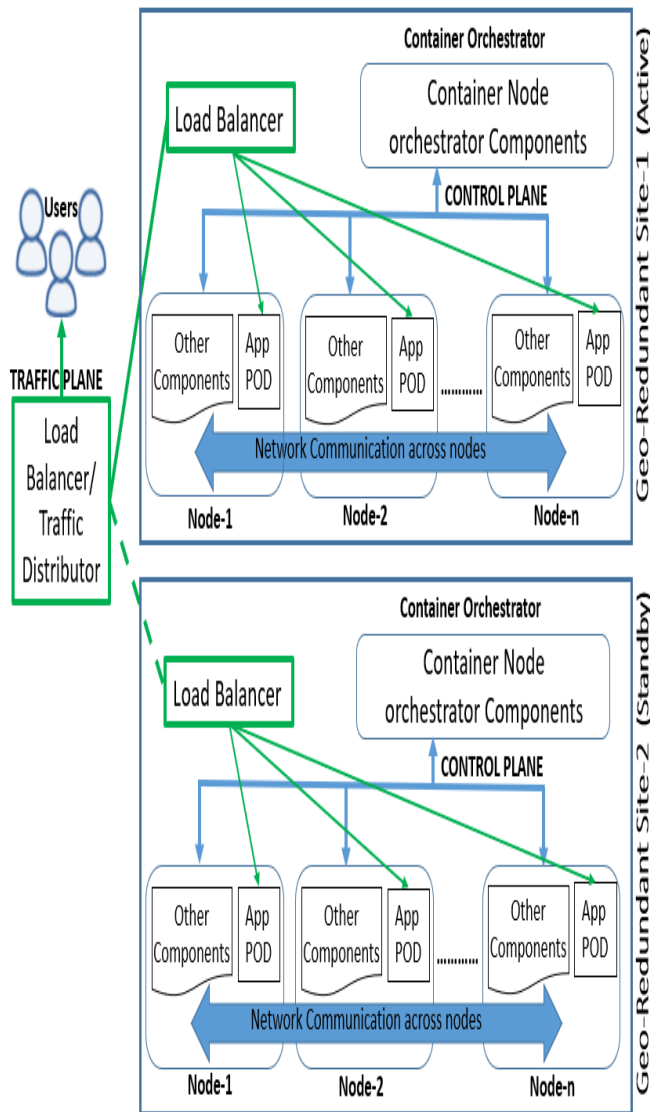


Fig. 3. HA and Geo-Redundant Setup

B. Testing Strategy

Our testing approach that has been formulated for the NFT of container-based applications comprises of 5 primary steps. Each step is executed for every Non-Functional characteristic that is being tested.

1) Build Environment –

Testing environment needs to be prepared that suits the characteristic under test. All pre-requisite to execute the intended test suite must be proactively analyzed and implemented in this phase.

2) Define Test Cases –

It is expected that there would be a list of predefined test cases to be mandatorily executed. Apart from this list, test cases for the characteristic needs to be prepared with expected results in conjunction with the application under test.

3) Execute Test Cases –

Execute the case for the application and observe behavior. Each approved test case from the previous step needs to be executed by confirming its pre and post test

case execution criteria. The results need to be documented post completion of the test case.

4) Collated Results and Logs –

Result of test case executed should be compared with expected result defined against each case. Collect all relevant logs related to the case which needs to be analyzed as part of testing procedure for any interim errors or warnings. This should be documented with possible workarounds and mitigation.

5) Analyze Logs and certify application –

Thoroughly analyze all logs to conclude the behavior of application in toto. A certification criterion should be generated which based on the complexity of the test cases and their results would certify the application into several categories.

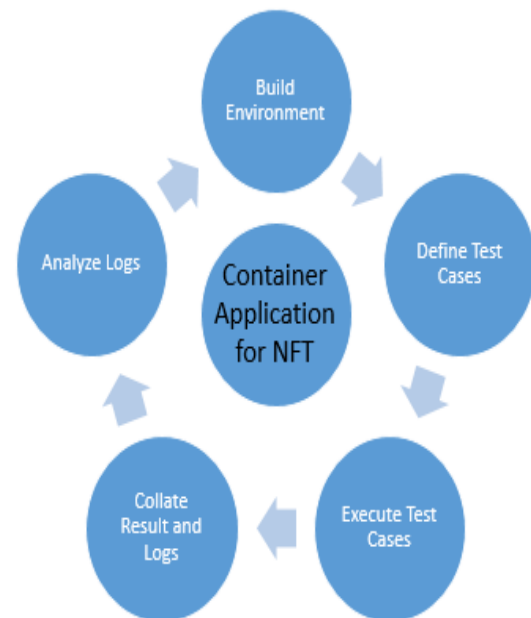


Fig. 4. Testing Strategy

IV. CONCLUSION

Container based applications are new and rapidly growing which makes the developers vulnerable to desist the mere building blocks of the application. There should be special focus on the non-functional aspect of container-based application because the performance, stability and robustness of the application is dependent on the environment it is hosted. With containers the underlying environment and hardware is completely abstracted and thus it is extremely important for container application developers to keep the non-functional characteristic of container-based application a priority, right from the design phase.

The proposed non-functional characteristics are primary building blocks of container-based applications. Before going production ready it is advised for any container-based application to test these non-functional characteristics. Proposed testing strategy would also help in streamlining the life cycle of testing where each test cycle for every non-functional characteristic will result into analysis of the outcome of the test. The proposed overall approach of testing and analyzing will help in enhancing the quality, stability,

robustness and availability of the application. Thus, improving the overall end user experience.

REFERENCES

- [1] T. Siddiqui, S. A. Siddiqui and N. A. Khan, "Comprehensive Analysis of Container Technology," 2019 4th International Conference on Information Systems and Computer Networks (ISCON), Mathura, India, 2019, pp. 218-223.
- [2] Kanso, Ali, Hai Huang, and Abdelouahed Gherbi. "Can linux containers clustering solutions offer high availability?" (2016).
- [3] "What is Non-Functional Testing?", <http://www.professionalqa.com/non-functional-testing>, (May 2019)..
- [4] Joe McKendrick, "Container technology market to grow 40% a year, analysts predict", <https://www.zdnet.com/article/container-market-keeps-growing/>, (17 January 2017).
- [5] David Linthicum, "3 best practices for container performance testing", <https://techbeacon.com/app-dev-testing/3-best-practices-container-performance-testing>.
- [6] Ingki, k ; Ari, I., Sozer, H:A; Survey of Software Testing in the cloud ;Software Security and Reliability Companion(SERE-C), Page:18-23, 2012.
- [7] WangJun, Fanpeng Meng; Software Testing Based on Cloud Computing; Internet Computing & Information Services(ICICIS), page:176-178, 2011.
- [8] Leah Riungu-Kalliosaari, Ossi Taipale, and Kari Smolander,; Testing in the Cloud: Exploring the Practice Computing & Processing, Page:46-51, 2012.
- [9] Amandeep Kaur, Navjeet Singh, Dr. Gurdev Singh; An overview of cloud testing as a service; International Journal of Computers & Technology, Volume 2 No. 2, April 2012.
- [10] Yatendra Singh Pundhir; Cloud Computing Applications and Their Testing Methodology; Bookman International Journal of Software Engineering, Vol. 2 No. 1, Jan-Feb-Mar 2013.
- [11] Scott Tilley, Tauhida Parveen; "Migrating Software Testing to the Cloud"; 26th IEEE International Conference on Software Maintenance in Timisoara, Romania, IEEE, 2010.
- [12] Y. Chen and Xh. Sun, "STAS: A Scalability Testing and Analysis System," in IEEE International Conference on Cluster Computing. Page:1-10, 25-28 Sept 2006.
- [13] Bernstein, David. "Containers and cloud: From lxc to docker to kubernetes." IEEE Cloud Computing 1.3 (2014): 81-84.
- [14] Tamanna Siddiqui, Riaz Ahmad, "A review on software testing approaches for cloud applications", Perspectives in Science (2016) (Elsevier) 8, 689—691.
- [15] Tamanna Siddiqui, Riaz Ahmad; Cloud Testing: A Systematic Review; International Research Journal of Engineering and Technology (IRJET); Volume: 02 Issue: 03 June-2015; e-ISSN: 2395 - 0056 p-ISSN: 2395-0072.
- [16] Dr. Tamanna Siddiqui and Riaz Ahmad, "Testing Framework for Geo-Redundant Cloud Based Applications" Journal of Advanced Research in Dynamical and Control Systems (JARDCS), Vol. 10, 04-Special Issue, 2018.