

HIGH VOLUME MEDIA SERVICE TESTING USING DEVOPS

Rakshitha M S¹, Vidya Raj C²

¹Student, MTech-Information Technology, The National Institute of Engineering, Mysuru-09

²Professor, Dept. of CS&E, The National Institute of Engineering, Mysuru-09

Abstract - This paper presents how the testing can be done on high volume data. It is done to check the data volume handled by the database. For testing we use a testing framework that uses a programming interface to the application to validate the behavior under test called API driven testing. The database is stretched to a threshold point by adding a large amount of data to it and then the system is tested for its response. REST Application Programming Interface (REST API) is a set of programming instructions for accessing a web-based software application or a set of commands used by an individual program to communicate with one another directly and use each other's functions to get information and that uses HTTP requests to GET, PUT, POST and DELETE data. CI/CD Pipeline implementation or the Continuous Integration/Continuous Deployment software is the backbone of the modern DevOps environment. The requirement of Continuous Integration & Continuous Deployment skills in various job roles such as Data Engineer, Cloud Architect, Data Scientist, etc. CI/CD bridges the gap between development and operations teams by automating build, test and deployment of applications. In this paper, we will know how the volume testing is done using automation testing.

Key Words: Automation testing, CI/CD pipeline, REST API, DevOps.

1. INTRODUCTION

Automation testing is a type of testing in which execution of test cases will be automated [1] and can automate some tedious but necessary tasks in a formalized testing process already in place or perform additional testing that would be difficult to do manually. Volume Testing is a type of non-functional testing, performing volume. Volume testing also called as flood testing is non-functional testing that is done to check the software or app for its performance against huge data of the database. It bypasses application user interface altogether, tested with a variety of input arguments to validate that the results that are returned are correct.

DevOps evokes an agile software development approach in an operational environment. Deployment is the period between the completion of the code by the developers and then placing of the code into normal production. Modern development organizations require entire teams of DevOps to automate and reduce the gap between the development team and the operation team. DevOps uses tools such as Git,

Docker, Jenkins. We have integrated all the tools to work together to host an application-based web server. And to test all our we have built a CI/CD pipeline.

Cloud platforms, such as Amazon, Azure and Ali, provide APIs for various services including infrastructure services, storage services, data services, and more and more rich application services such as cognitive services and machine learning services [4].

The high-volume data will run on the different cloud services and is offered as a Software as a Service (SaaS). It is built from the ground up leveraging Agile methodology, bringing rapid innovation, and new monetization opportunities to operators, cloud services enable IPTV Multicast (RTP) linear video delivery to Managed Set-top boxes and Adaptive Bit Rate (ABR) linear video delivery to reach clients that include iOS and Android devices and Unmanaged (OTT) set-top boxes. It is a TV Platform Service also provides a unified storefront and a user experience (UX) for delivering ABR video on demand for set-top boxes and we will be seeing how the high-volume data are automated using automation testing.

2. RELATED WORK

[1] Jingfan Tang and Xiaohua Cao, they have introduced a framework called adaptive framework keyword-driven automation testing, which is used to convert keyword-based test cases into different kinds of test scripts automatically and to be executed by different test applications under different test environments

[2] Dietmar Winkler, Reinhard Hametner, Stefan Biffel, they have introduced automation component, which is for effective unit testing, the design of a test-driven automation (TDA) component with separate aspects aimed at automation, diagnosis, and testing to improve testability.

[3] Dietmar Winkler, Reinhard Hametner, Thomas Östreicher, Stefan Biffel, have established a framework called Test-First Development (TFD) which is used in IT software development - promises to support test automation in automation systems development, helps to improve automation systems development and product quality.

[4] Junyi Wang, Xiaoying Bai, Haoran Ma, Linyi Li, Zhicheng Ji, ATcloud, a model-based automatic testing framework and a prototype system to support Cloud API testing, it includes approaches like API Understanding, Test Generation, Test Engine and Test Analyzer, which is helpful in API testing.

[5] Andrea Arcuri Westerdals Oslo ACT, Oslo, Norway and SnT, The RESTful web service will provide data via an API over the network using HTTP, possibly interacting with databases and other web services. Testing a RESTful API poses challenges, as inputs/outputs are sequences of HTTP requests/responses to a remote server and consider testing from the point of view of the developers, which do have full access to the code that they are writing and a novel approach that can automatically generate integration tests for RESTful web services.

3. PROBLEM STATEMENT

We have large number of API's which needs to be tested in regular basis, from the time the development team gives us the code and commits it to the time we get it into production, so we automated testing scripts and placed it in pipeline and executed in daily basis, after every execution we make sure everything is green and also update failed scripts.

3. TOOLS

3.1 DevOps

DevOps is a software development approach which involves continuous development, continuous testing, continuous integration, continuous deployment and continuous monitoring of the software throughout its development life cycle. This is exactly the process adopted by all the top companies to develop high-quality software and shorter development life cycles, resulting in greater customer satisfaction, something that every company wants. Steps in DevOps are shown in Fig1.

1. Coding – code development and review, source code management tools, code merging.
2. Building – continuous integration tools, build status.
3. Testing – continuous testing tools that provide quick and timely feedback on business risks.
4. Packaging – artifact repository, application pre-deployment staging.
5. Releasing – change management, release approvals, release automation.
6. Configuring – infrastructure configuration and management, infrastructure as code tools.
7. Monitoring – applications performance monitoring, end-user experience.

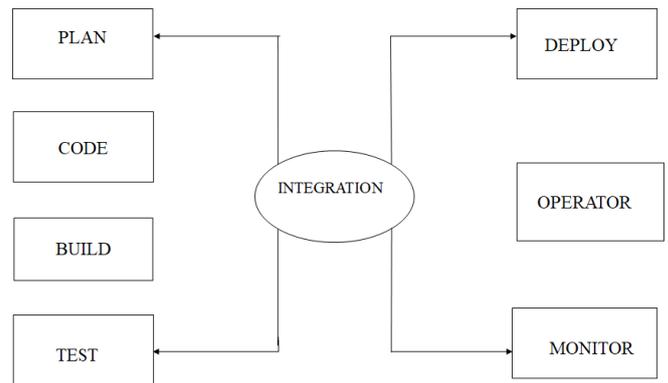


Fig-1: Steps in Devops

3.2 REST API

REST stands for Representational State Transfer. It means when a RESTful API is called, the server will transfer to the client a representation of the state of the requested resource. A RESTful web application exposes information about itself in the form of information about its resources [5].

There are mainly 4 methods involve in API Testing like GET, POST, Delete, and PUT.

- *GET*- The GET method is used to extract information from the given server using a given URI. While using GET request, it should only extract data and should have no other effect on the data.
- *POST*- A POST request is used to create a new entity. It can also be used to send data to the server, for example, customer information, file upload, etc. using HTML forms.
- *PUT*- Create a new entity or update an existing one.
- *DELETE*- Removes all current representations of the target resource given by a URI.

3.3 GIT

A Version Control System (VCS) is the management used to change any kind of document but it is used to keep track of changes in source code during software development. There are two type of Version control system: Centralized VCS and distributed VCS. A centralized version control system is a client-server approach. The client checks out a copy from the centralized repository. All these data and history is stored and retrieved from the central repository. On the other hand, we have the distributed Version control system which is a peer to peer approach. Network connection is needed when it is to be shared. Workflow of git is defined in Fig 2.

By using these git commands pull requests are created

1. Branch name - users/ {your name}/ {bug number}
2. git checkout -b {branch name} origin/dev/master

3. Setenv
4. Git add.
5. git localbuild
6. git commit -m "{comment}" -m "#bug_number"
7. git push origin {branch name}

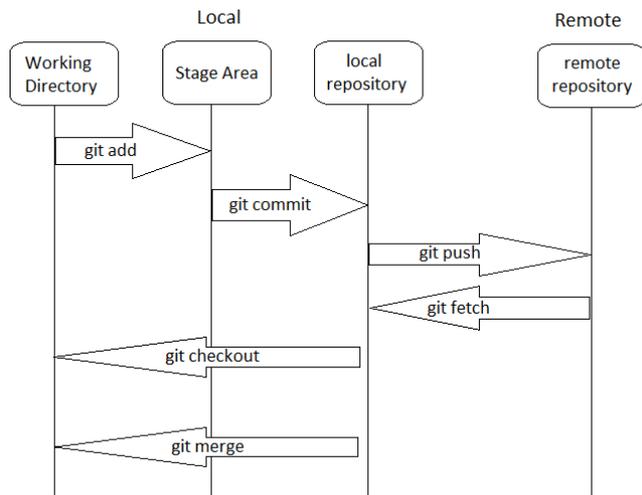


Fig-2: workflow of git

Version controlling our code provides several benefits:

- 1) The ability to work on the same project by more than one person at the same time. Each person will have a copy of the source code to do whatever changes required, then changes are merged with the main code, after the necessary approvals.
- 2) The history stored by a version control system would allow a developer to examine changes to faulty code.
- 3) The code can be branched. Developers can work on separate branches of the source code to create and test new features. In the end, a branch can be merged with the main branch (called master)

3.4 CD\CI

Continuous Integration (CI) is a development practice that requires users to integrate code into a shared repository several times a day. Continuous Integration is achieved with build servers which enables building certain tasks whenever there is a commit for a feature. Each check-in triggers an automated build, allowing teams to detect problems early. Continuous integration approach through Jenkins. Continuous delivery is a software engineering practice where team producing software as feature in small cycles are continuously deployed and ensures that it could be released for production. Continuous delivery will fit right into the big picture of the software development life cycle

with a major role to play in the delivery of the software to the production environment.

Continuous Deployment: Over here, full automation is achieved. Code is deployed by the application code or to the binary artifacts and is given to the live production servers to be consumed by end users. CI/CD makes use of several tools including build server such as Jenkins, repository for fetching the builds and storing the artifacts for reporting. Continuous delivery will make use of deployment tool which would communicate will build server tool.

3.5 JENKINS

Jenkins is a widely used continuous integration tool available in open source. Jenkins has all basic continuous integration services for software development. It supports version control system tools including Git, Mercurial, and can execute Apache Ant and Apache Maven based projects as well as arbitrary shell scripts and Windows batch commands. Jenkins stores output of end to end automation and helps to fix the problem by providing dashboard plugins.

3.6 DOCKER

The Modern Platform for High-Velocity Innovation. The only independent container platform that enables to seamlessly build, share and run any application, anywhere from hybrid cloud to the edge. Using these technologies and other useful methods and techniques we would able to build, monitor and maintain the high-volume media data and the CI pipeline.

4. OVERVIEW OF TESTING

An environment should be built before demonstrating DevOps, after writing a code for the specified requirement. Git is used for source-code management, Docker is used to create a container inside which the application will be executed. Jenkins is used to build, test and deploy software, centralizes the testing environment by building CI/CD pipeline which includes auto build, auto promote etc. Using ADO, the testcases written will be reviewed and merged in a git repository by creating a pull request. It will let team to review code, comment and give feedback on changes before merging into the master branch. Reviewers can go through the proposed changes, leave comments, and vote to approve or reject the code.

The procedure as follows:

- The pull request can be viewed in a specific repository in a project, got to that project in the web portal and select Repos > Pull requests.
- After pushing a branch, Azure Repos prompts to create a pull request. This prompt is displayed on Pull Requests and Files.

- Select Create a pull request to go to a page where you can enter your pull request details and create the pull request.
- Add detail to your pull request After creating a pull request, add pull request title, add a detailed description, add reviewers, link work items, and make comments to explain the changes. When you're ready to create the pull request and have your changes reviewed, select Create.
- Reviewer will add comments to the pull request to make suggestions, reply to previous comments, and point out problems with the projected changes.
- Update comment status to let reviewers know what you are doing to address the concerns brought up in their review. New comments start in Active status. Select Resolve or Reply & resolve to update comments in the conversation.

[3] Dietmar Winkler, Reinhard Hametner, Thomas Östreicher, Stefan Biff1" A Framework for Automated Testing of Automation Systems".

[4] Junyi Wang, Xiaoying Bai, HaoranMa, Linyi Li, Zhicheng Ji," Poster: Cloud API Testing",10th IEEE International Conference.

[5] Andrea Arcuri Westerdals Oslo ACT, Oslo, Norway and SnT, "RESTful API Automated Testcase Generation", IEEE International Conference 2017.

Vote on the changes

- Approve: Agree with the proposed changes in the pull request as-is.
- Approve with suggestions: Agree with the pull request but provide optional suggestions to improve the code.
- Wait for author: Don't approve the changes and ask the author to review your comments. The author should let you know to review the code again after they address your concerns.
- Reject: The changes aren't acceptable. If you vote this way, leave a comment in the pull request to explain why.
- Reset feedback: Choose Reset feedback to remove your vote.

The number of required approvals in a pull request can be set from the branch policy for the branch. A pull request that meets the required number of approvals can be completed and after getting the approval the code can be merged to the master. After finishing all the testing testcase will be automated and made run CI\CD pipeline.

5. CONCLUSION

In this paper, we saw how the high-volume service can be done using different tools, code-and-fix methods in software development of systems engineering hinder efficient progress, maintenance, processes and technology for an automated software delivery that is agile, scalable and cost-effective.

REFERENCES

- [1] Jingfan Tang and Xiaohua Cao, Street Technology (Zhejiang) Co. Ltd "Towards Adaptive Framework of Keyword Driven Automation Testing", IEEE Conference 2008.
- [2] Dietmar Winkler, Reinhard Hametner, StefanBiff1, "Automation Component Aspects for Efficient Unit Testing".