E2E Automation from JENKINS to LISA

Project: CLIENT_MICROSERVICE

Ram Aurovind

Abstract: The CLIENT_MICROSERVICE Trade Tools team's initiative, E2EAUTOMATION, stands out as a forward-thinking response to the persistent inefficiencies in international CRM, SRM, shipping, and banking systems. Rather than following traditional, error-prone methods, the project proposes an intelligent, microservices-driven platform capable of classifying commodities and accurately predicting duties, taxes, and transit times all in just one click. This isn't merely a technical shift; it signals a deeper movement toward data democratization and operational agility. The fusion of Big Data pipelines, machine learning, and parameterized Jenkins pipelines has streamlined workflows, drastically cut down manual coding and approvals, and introduced reusable, modular architectures. What's particularly noteworthy is how the team tackled real pain points fragmented development, delayed testing, and complex deployments by consolidating job executions, reducing pipeline code by over 60%, and implementing a shared Jenkins library for scalable, consistent integrations. This suggests that E2EAUTOMATION isn't just about code efficiency; it's about reimagining how automation can elevate the entire user journey, from back-end logic to real-time decision-making. It is evident that this architecture offers a replicable blueprint for any enterprise aiming to modernize their trade and logistics operations while ensuring traceability, governance, and rapid deployment.

Keywords: microservices automation, intelligent pipeline, Jenkins shared library, Big Data integration, commodity classification

1. Introduction

CLIENT_MICROSERVICE Trade Tools team began a groundbreaking new initiative named E2EAUTOMATION to simplify the international CRM / SRM / SHIPPING / BANKING experience for CLIENT_MICROSERVICE customers. So have to build an innovative, intelligent platform that classifies commodities to a fully qualified code with a single click.

By applying Big Data, Micro services and data science techniques to historical shipment and tariff data, we were able to deliver a simplified, streamlined customer experience with more accurate estimated duties, taxes and transit time Any Operating company looking for accurate commodity classification and accurate duties, tax and transit and also other non-clearance and brokerage platforms can leverage Trade Tools intelligent micro services

E2EAUTOMATION involves innovative idea of improving the customers experience by providing accurate product code, estimated duties, estimated taxes, transit time, and digitalized experience by utilizing the cutting-edge technologies like Big Data and Analytics. To be responsible for ingesting data from various data sources into the Big Data Lake, preparing Data engineering pipelines and machine learning models to create datasets which will be used to render data to Business services

2. Summary

• A Jenkins file is a text file that is written using Groovy syntax for defining

- Continuous Integration, Continuous Deployment and Continuous Testing (CI/CD/CT) pipeline as Code.
- supporting version control, code review and automation of the build, test and deployment process within Jenkins.
- It encapsulates the importance of pipelines configuration for providing consistency and repeatability across multiple software development projects.

3. Pre-Requisites

- Jenkins : https://jenkins.ENV(dev/stage/prod).cloud.customer.com :8443/jenkins/job/CLIENT_MICROSERVICE/job/Micro Services/
- LISA Server: c0008666.ENV(dev/stage/prod).cloud.customer.com

4. Pipeline

Pipeline as a code

5. Location

- Common for all services: https://jenkins.ENV(ENV(dev/stage/prod)/stage/prod).cl oud.customer.com:8443/jenkins/job/CLIENT_MICROS ERVICE/job/MicroServices/job/Scripted_Microservice/j ob/
- Below is an example for HSValidation path: https://jenkins.ENV(ENV(dev/stage/prod)/stage/prod).cl oud.customer.com:8443/jenkins/job/CLIENT_MICROS ERVICE/job/MicroServices/job/Scripted_Microservice/j ob/HSV_DSL/

Volume 14 Issue 6, June 2025 Fully Refereed | Open Access | Double Blind Peer Reviewed Journal www.ijsr.net

International Journal of Science and Research (IJSR)

ISSN: 2319-7064

Impact Factor 2024: 7.101

| ┢ Up | Pipeline HSV_DSI |
|-----------------------|--|
| 🔍 Status | |
| P Changes | This build requires parameters: |
| Build with Parameters | ENV_NAME DEPLOY_L1_L2_L3 SELECT ENV TO DEPLOY |
| Delete Pipeline | APP_NAME HSVALIDATION_BUSINESS_SERVICE V |
| 🏇 Configure | SELECT APP TO DEPLOY |
| Move/Copy/Promote | Build |
| 🗞 Skip Builds | |

6. Current Problem Statement Faced in the Project

- Lack of a version control system, resulting in developers writing and manually integrating code independently.
- Slow and ineffective testing due to manual execution.
- Limited collaboration resulting from developers working on completely separate code branches.
- Infrequent integrations and testing that introduce significant risks during software releases.
- Manual deployments and Approvals from the Loadmaster (Sr. Project manager) across development, staging, and production environments.

7. Proposed Solution

In order to remove the manual job creation and management, Pipeline **as Code with** a parameterized **build** has been proposed and implemented

It allows the Release Team and Developers to run parametrized pipeline job based upon the environment's processes which will be stored as code (Jenkins File), stored and versioned in a source repository

8. Pipeline Summary and Execution

- Parametrized build will help customizing the build and deployment process with minimum code
- Single Job Triggered across Multiple Environments in 3 levels (L1-Dev1, L2-Dev2 and L3-Stage1) in a single stage, (L4-STAGE2, L6-PROD) in 2 different stages as shown below
- Application components under HS-Validation in git lab, which was created as a separate pipeline has been merged into a single pipeline where in it will execute the code based upon the selection of Application component parameter
- please find below
- Pipeline code has been decreased more than 60% lines of code as compared to the existing code

- Stages are rather executed based on the condition/parameter
- We can use Jenkins shared library which will give more benefits across the projects, Single shared library will be access all the projects.
- Through Shared library Jenkins file/Pipeline code will decrease a lot, only reference of the shared library will be given in pipeline code

9. Pipeline Details

- 1) Single parameterized based on ENV and Application
- 2) Pipeline runtime based on stages
- 3) Now based one the conditions/approvals we will be running/executing stages with proper email notifications
- No. of lines of Pipeline as a code decreased less than 50% from previous code
- 5) Multiple jobs merged to single job with parameterized build
 - a) Now the pipeline code is completely modularized with specific method
 - b) Can send build log after each and every stage completion for Load Master Approval
- 6) Load master Can approve the pipeline by Email or through Pipeline stage
- 7) Single method is used for multiple activities with the parameter at run time
- 8) Easy to Debug the Pipeline stages
- 9) Each and every stage can be viewed across pipeline
- 10) The method used is Single Build Multiple deployments
- 11) Build and deploy the code in L1 and deploy the same code across L2, L3, L4 and L6

Note: Below is the detailed description of each pipeline stage

10. Screenshots of pipeline stages (for HS Validation) is the Name of the Micro Service Application

• Single parameterized based on ENV and Application

Volume 14 Issue 6, June 2025 Fully Refereed | Open Access | Double Blind Peer Reviewed Journal www.ijsr.net

International Journal of Science and Research (IJSR)

ISSN: 2319-7064

Impact Factor 2024: 7.101



Selection of Application and Env for Approval

| ▶ Up | Pipeline HSV DSL | | | | |
|-----------------------|--|--|--|--|--|
| 🔍 Status | · · · · · · · · · · · · · · · · · · · | | | | |
| Changes | i his build requires parameters: | | | | |
| Build with Parameters | ENV_NAME DEPLOY_L1_L2_L3 SELECT ENV TO DEPLOY | | | | |
| 🔰 Delete Pipeline | APP_NAME HSVALIDATION_BUSINESS_SERVICE V | | | | |
| 🐡 Configure | HSVALIDATION_BUSINESS_SERVICE | | | | |
| Move/Copy/Promote | Build HSVALIDATION_DATA_SERVICE | | | | |
| lefter Skip Builds | | | | | |

• Selection of ENV starting with DEPLOY_L1_L2_L3, Same code built in L1 will be deployed across L2 and L3 for based upon LOAD MASTER Approval

| | | | | | | | | | | | | <u> </u> | niarge <u>contigure</u> | |
|------------------|----------------|-----------------|--------------|--------------|-------------------|------------------|-----------------------|-------------------|--------------|-----------------------|-------------------|--------------|-------------------------|-------------------|
| Code checkout | maven build | Creating TAR | CodeAnalysis | Deploy L1 | smoke Ttesting | Smoke Results | Email Notification | Email Approval | Deploy L2 | Email Notification | Email Approval | Deploy L3 | Email Notification | Email Approval |
| 2s | 1min 1s | 593ms | 2s | 4min 15s | 1min 50s | 634ms | 497ms | 842ms | 55s | NaNy NaNd | NaNy NaNd | 2min 57s | NaNy NaNd | NaNy NaNd |
| 5s | 1min 54s | 504ms | 85 | 5min 13s | 40s | 594ms | 448ms | (passion) 19mm | | | | | | |
| almost comp | lete | | | | | | | Success | | | | | | |

• Load Master receives email with Build Logs and Build approval

| Build I | Notification: CASCADE/MicroServices/Scripted_Microservice/HSV_DSL-Build# | |
|---------|---|--|
| N | no-reply@corp.ds tedex.com Today, 3:01 PM Aurovind Chakravarthi (QSV) × | |
| | build.zip 16 KB | |
| | Download | |
| | CASCADE / MicroServices / Scripted_Microservice / HSV_DSL-21 | |

• Load Master validates the build logs and Click on the email approval Link

Volume 14 Issue 6, June 2025 Fully Refereed | Open Access | Double Blind Peer Reviewed Journal www.ijsr.net

Paper ID: SR25529234731

International Journal of Science and Research (IJSR) ISSN: 2319-7064 Impact Factor 2024: 7.101

| Inbox | Filter 👻 | LoadMaster approval and Submit Results |
|--|---|---|
| Next: Synter Orshore Team 1 Tom no-reply@corp.do Learthfaster approved as a more the log Montemary Surged | m m Minister 301 PM | No-reply@corp.d: Toda, 3.72 Aurono |
| no-reply@corp.ds Build NonReadow (Mosterword Sciented, March no-reply@corp.ds | m li iervicev 3-01 PM roservice MSV | Job MicroServices/Scripted_MicroService/HSV_D6LD1F_https://nam03.safelinks.protection.outlook.com/2 url-am/05/SAR52P3/2Fjerkins.erod.cloud.fedex.com/53884435/2Fjerkins5/2Fjerki2FCASCADE3/2Fjbb/3FN/croServices352Fjob/32FScripted_Microsen.se/s2Fjob/s2FHSV_D5L352E213/2Fjop url/samp.data.e/2fs/CO15/7Caurovind.chakravathi.os/%40/tedex.com/57Cf4483c14c27b42?ezced58d721bb6aa6357Cb/45c813dcef41884575a12c2h15effo7C15/7C63701486114059 22658amp.sdata.e/Xm7y81496Lp5wCV252F3100M2I575y00M5c6KS82R8hY53D8Aamp.teserved.e/ |

• Link in email will take the user to following Jenkins page

| Have the load | Imasterapproval passed? |
|-------------------------|--|
| loadMasterValidation | |
| loadMasterApproveResult | Load Master/Validates Passed What is the overall result? |
| Proceed Abort | |

• Load Master gets the following options to Pass or Fail the builds

| Have the load | Imasterapproval passed? | | | |
|-------------------------|--|--|--|--|
| loadMasterValidation | | | | |
| loadMasterApproveResult | Load Master/validates Passed Passed Passed | | | |
| Proceed Abort | Failed | | | |
| | Passed with Concessions | | | |

• If the Load Master Select Failed, the build will get failed and email will be sent to the stake holders

0

| Input requested |
|---|
| Approved by <u>Aurovind Chakravanthi (O</u> |
| [Pipeline] echo |
| Load MasterValidates: failed |
| [Pipeline] echo |
| Overall Manual Test Result: Failed |
| [Pipeline] } |
| [Pipeline] // timeout |
| [Pipeline] } |
| [Pipeline] // script |
| [Pipeline] } |
| [Pipeline] // stage |
| [Pipeline] stage |
| [Pipeline] { (Deploy L3) |
| [Pipeline] script |
| [Pipeline] { |
| [Pipeline] withCredentials |
| [Pipeline] { |
| [Pipeline] step |
| [Pipeline] } |
| [Pipeline] // withCredentials |
| [Pipeline] } |
| [Pipeline] // script |
| [Pipeline] } |
| [Pipeline] // stage |

If the build logs along with ENV(dev/stage/prod) cases are passed, Load Master should approve for deployment to next level. The same process is repeated for remaining levels too.

Volume 14 Issue 6, June 2025 Fully Refereed | Open Access | Double Blind Peer Reviewed Journal www.ijsr.net

•

Timeout set to expire in 1 hr 0 min [Pipeline] { [Pipeline] mail [Pipeline] input Input requested Approved by Aurovind [Pipeline] echo Load MasterValidates: pass [Pipeline] echo Overall Manual Test Result: Passed [Pipeline] } [Pipeline] // timeout [Pipeline] } [Pipeline] // script [Pipeline] } [Pipeline] // stage [Pipeline] stage [Pipeline] { (Deploy L2) [Pipeline] script [Pipeline] { [Pipeline] withCredentials [Pipeline] { [Pipeline] step Running job as alternative user '****'. Deploying component versions '{HSVALIDATION_DATA_SERVICE=[Jenkins.Pipeline.20]}' Starting deployment process 'Deploy_L1_L2_L3' of application 'HSVALIDATION_SERVICE' in environment 'L2' Deployment request id is: '16c96a3a-ebee-fc23-db5d-b405511af4a8' Deployment is running. Waiting for UCD Server feedback.

11. Automation Testing-CA Lisa-CI CD CT Micro Service Implementation Guide



12. Conclusion

- The Pipeline as code is a flexible and scalable framework for implementing CI/CD/CT workflows.
- By implementing best practices for organizations, we are able to accelerate the delivery cycles, mitigate risks and improve the quality of the software, optimize internal processes,
- And it did eradicate human error. we are able to resolve the issues and increase the productivity by doing CI/CD/CT without any major Outage across Multi Environments.

Volume 14 Issue 6, June 2025 Fully Refereed | Open Access | Double Blind Peer Reviewed Journal <u>www.ijsr.net</u>