

#### National College of Ireland

#### **Project Submission Sheet**

Student Name:	EZEKIEL ADEDAYO AYANDA					
Student ID:	23129522					
Programme:	MSc Cloud Computing	Year:	2024			
Module:	Configuration Manual					
Lecturer:	Dr. Rashid Mijumbi					
Submission Due Date:	Wednesday, 29th January 2025					
Project Title:	Scalability Optimization in Identity Mana Native Applications	agement Sys	stems for Cloud-			
Word Count:						

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

<u>ALL</u> internet material must be referenced in the references section. Students are encouraged to use the Harvard Referencing Standard supplied by the Library. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action. Students may be required to undergo a viva (oral examination) if there is suspicion about the validity of their submitted work.

Signature:Ezekiel AyandaDate:27/01/2024

#### PLEASE READ THE FOLLOWING INSTRUCTIONS:

- 1. Please attach a completed copy of this sheet to each project (including multiple copies).
- 2. Projects should be submitted to your Programme Coordinator.
- 3. You must ensure that you retain a HARD COPY of ALL projects, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer. Please do not bind projects or place in covers unless specifically requested.
- 4. You must ensure that all projects are submitted to your Programme Coordinator on or before the required submission date. Late submissions will incur penalties.
- 5. All projects must be submitted and passed in order to successfully complete the year. **Any project/assignment not submitted will be marked as a fail.**

Signature:	
Date:	
Penalty Applied (if applicable):	



# Configuration Manual

MSc Research Project Cloud Computing

Ezekiel Ayanda Student ID: 23129522

School of Computing National College of Ireland

Supervisor: Rashid Mijumbi

#### National College of Ireland Project Submission Sheet School of Computing



Student Name:	Ezekiel Ayanda
Student ID:	23129522
Programme:	Cloud Computing
Year:	2024
Module:	MSc Research Project
Supervisor:	Rashid Mijumbi
Submission Due Date:	12/12/2024
Project Title:	Configuration Manual
Word Count:	1835
Page Count:	22

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

<u>ALL</u> internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Ezekiel Ayanda
Date:	10th December 2024

#### PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).Attach a Moodle submission receipt of the online project submission, to<br/>each project (including multiple copies).You must ensure that you retain a HARD COPY of the project, both for

your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

# Configuration Manual

Ezekiel Ayanda 23129522

# 1 Introduction

This document serves as a comprehensive guide to perform a successful reproduction and configuration of the tools and technology utilized in the course of implementing this project with reference to the code artifact and online documentation were necessary. Considerate effort has been put into calling out possible issues encountered with setup and remediation steps taken to easy next reproduction of steps.

# Contents

1	Introduction	1
<b>2</b>	Setup Prerequisites	1
3	Azure Kubernetes Cluster	<b>2</b>
4	Java SpringBoot Application	3
<b>5</b>	Identity Management System	6
6	Time Series Monitoring	11
7	Load Testing	17

# 2 Setup Prerequisites

For the scope of this project, certain versions of tools and platforms have been utilized. While it might not be mandatory, it is recommended that the same versions are used were necessary for best results and to avoid configuration mishaps.

Component	Type/Version
Compute Instance	Virtual Machine Scale Sets
Operating System	Ubuntu Linux
Orchestration Tool	Kubernetes (Version 1.29.9)
Container Tool	ContainerD (Version 1.6.26)

Table 1: Azure Kubernetes Cluster

vCPU	2 vcores
Memory	7 GB
Node VM Size	$Standard_D S2_v 2$
OS Type	Ubuntu Linux
Cost of services	\$0.146/Hour

Table 2: VMSS / Node Specification

# 3 Azure Kubernetes Cluster

As a prerequisite to all configurations in this project, it is expected that you already have at least contributor access to the Azure Kubernetes cluster.

#### 3.1 Install AZ CLI

Azure CLI is available for download and installation for any of your preferred environments (Windows, macOS and Linux environments). This can be accessed at Azure CLI Installation Guide

# 3.2 Deploy AKS Cluster

Sign into Azure CLI Use the following command to sign into your Azure account: az login

**Set Subscription Context** Set the subscription context to your subscription:

```
az account set --subscription <sub name or id>
```

Create AKS Cluster Run the following command to create the AKS cluster:

```
az aks create \
--resource-group <resource-group-name> \
--name <aks-cluster-name> \
--node-count <number-of-nodes> \
--node-vm-size <vm-size> \
--generate-ssh-keys \
--kubernetes-version 1.29.9 \
--network-plugin azure
```

# 3.3 Install Kubectl, Docker and Helm Chart

Most of the deployments in the cluster are deployed using Helm as the package manager. Docker is needed for build the Java app and Kubectl is required to run commands against cluster. Official guides for installation of these can be followed for installation based on Operating system.

- Kubectl: https://kubernetes.io/docs/tasks/tools/
- **Docker**: https://docs.docker.com/engine/install/

#### • Helm: https://helm.sh/docs/intro/install/

#### 3.4 Nodes and Namespace configuration

Create dedicated nodes for different parts of the cluster along with corresponding namespaces. Three user node pools can be created alongside three namespaces. The system node pool is specifically reserved for native Kubernetes operational workloads.

Node pool	Namespace
appnode1	appnode
idmsnode1	idmsnode
monitoring	monitoring

Table 3: Cluster workload Recommended Mappings

Defining scoped nodes and namespaces ensures proper grouping of workloads within the cluster, leveraging node selectors and namespace parameters for deployment management. User nodes can be created via kubectl or via the Azure portal using the system specifications in 2

# 4 Java SpringBoot Application

#### 4.1 Integration and setup

Configuration prerequisites include:

- Realm Name
- Client ID
- Client Secret

Include the necessary dependencies for OAuth2 and Keycloak in your pom.xml:

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-oauth2-client</artifactId>
</dependency>
    <dependency>
        <groupId>org.keycloak</groupId>
        <artifactId>keycloak-spring-boot-starter</artifactId>
        <version>26.0.6</version>
</dependency>
```

Create a security configuration class to enable OAuth2-based authentication. This handles the connectivity to Keycloak and redirects users for login in /Java Springboot App/src/main/java/co

```
@Configuration
@EnableWebSecurity
public class SecurityConfig {
    @Bean
    public SecurityFilterChain securityFilterChain(HttpSecurity
       http) throws Exception {
        http
            .cors(AbstractHttpConfigurer::disable)
            .csrf(AbstractHttpConfigurer::disable)
            .authorizeHttpRequests(req -> req
                 .requestMatchers(AntPathRequestMatcher.antMatcher
                   ("/**")).permitAll()
                .anyRequest()
                .authenticated()
            )
            .sessionManagement(session -> session
                .sessionCreationPolicy(SessionCreationPolicy.
                   STATELESS));
        return http.build();
    }
}
```

#### 4.2 Push App to Docker or Azure Container Registry

Build the Docker image docker build -t <your-dockerhub-username>/java-spring-app:v1 .

Push the Docker image to a container registry

docker push <your-dockerhub-username>/java-spring-app:v1

Creating the Application in the Kubernetes Cluster

The Deployment manifest file can be found in /yamls/java-app-deployment.yml. Update your docker image and selector. Also set Environment variable values to match Keycloak deployment.

```
apiVersion: apps/v1
kind: Deployment
metadata:
    name: java-app
    labels:
        app: java-app
spec:
    replicas: 2
    selector:
```

```
matchLabels:
   app: java-app
template:
 metadata:
   labels:
     app: java-app
 spec:
   containers:
   - name: java-app
     image: <your-dockerhub-username>/java-spring-app:v1
     ports:
     - containerPort: 8080
     env:
     - name: iam.integration.keycloak.url
       value: "http://<keycloak-host>/"
     - name: iam.integration.keycloak.client-id
       value: "<ClientID>"
     - name: iam.integration.keycloak.realm
       value: "<realm name>"
     - name: iam.integration.keycloak.username
       value: "admin"
     - name: iam.integration.keycloak.password
       value: "adminpassword"
     - name: iam.integration.keycloak.client-secret
```

The Service manifest file can be found in /yamls/java-app-deployment.yml

```
apiVersion: v1
kind: Service
metadata:
   name: java-app-service
spec:
   selector:
    app: java-app
   ports:
        - protocol: TCP
        port: 80
        targetPort: 8080
   type: LoadBalancer
```

Confirm Java App Pod and Service in AKS

kubectl get pods -n appnode
kubectl get svc -n appnode

# 5 Identity Management System

# 5.1 Keycloak Installation

This configuration should get Keycloak up and running for a development use case. A full list of Bitnami Helm chart parameters for Keycloak can be found on **Bitnami's GitHub** repository.

```
helm install keycloak bitnami/keycloak \
--namespace idmsnode \setminus
--set service.type=LoadBalancer \
--set auth.adminUser=admin \
--set auth.adminPassword=<securepassword> \
--set nodeSelector.agentpool=idmsnode1 \
--set replicaCount=3 \
--set persistence.enabled=true \
--set persistence.size=10Gi \
--set postgresql.enabled=true \
--set postgresql.postgresqlUsername=admin \
--set postgresql.postgresqlPassword=<securepassword> \
--set postgresql.postgresqlDatabase=keycloak \
--set resources.requests.cpu=500m \
--set resources.requests.memory=1Gi \
--set resources.limits.cpu=1 \
--set resources.limits.memory=2Gi
```

Confirm the pods, Service and Persistent Volume Claim deployment in AKS

kubectl get pods -n idmsnode
kubectl get svc -n idmsnode
kubectl get pvc -n idmsnode

[ezekielayanda@Ezekiels-⊬	lacBook-	Air bin 9	6 kubectl get	pods -n idmsnod	e					
NAME	READY	STATUS	RESTARTS	AGE						
keycloak-0	1/1	Running	0	13h						
keycloak-1	1/1	Running	0	13h						
keycloak-postgresql-0	1/1	Running	0	13h						
[ezekielayanda@Ezekiels-M	lacBook-	Air bin 🤋	6 kubectl get	svc -n idmsnode						
NAME	TYPE		CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE				
keycloak	LoadBa	lancer	10.0.182.210	4.209.42.13	80:30811/TCF	P 17d				
keycloak-headless	Cluste:	rIP	None	<none></none>	8080/TCP	17d				
keycloak-postgresql	Cluste:	rIP	10.0.28.88	<none></none>	5432/TCP	17d				
keycloak-postgresql-hl	Cluste:	rIP	None	<none></none>	5432/TCP	17d				
[ezekielayanda@Ezekiels-M	ezekielayanda@Ezekiels-MacBook-Air bin % kubectl get pvc -n idmsnode									
NAME	ST	ATUS VO	DLUME			CAPACITY	ACCESS MODES	STORAGECLASS	VOLUMEATTRIBUTESCLASS	AGE
data-keycloak-postgresq	L-0 Bo	und pv	/c-06d7d6c5-d	bb0-4fcf-8900-ca	73c531848b	8Gi	RWO	default	<unset></unset>	17d
ezekielayanda@Ezekiels-M	lacBook-	Air bin 🤋	6 [							

Figure 1: Keycloak pods, services and Persistent Data Volume for Postgresql

Obtain and verify Keycloak admin password

```
kubectl get secret keycloak -n idmsnode -o jsonpath="{.data.admin-password}"
| base64 --decode
```

Keycloak Admin console can be accessed over the internet with the External IP or via port Forwarding.

> C	D	A Not Secure	$\textbf{4.209.42.13} / \texttt{realms/master/protocol/openid-connect/auth?client\_id=\texttt{security-admin-console&redirect\_u}$	Ů  Ÿ	ඩ් 🕹 🗖 🐔 Update
			<b>KEYCLOAK</b>		
			Sign in to your account Username or email Password Password		
			Sign In		

Figure 2: Keycloak Admin Console Login page

Enable Autoscaler for Keycloak which will be working with Infinspan to on-boarding new instances into the cache cluster. (Minimum replicacount to enable Keycloak clustering is 2). Update this in the value.yaml file

```
autoscaling:
enabled: true
minReplicas: 2
maxReplicas: 3
targetCPU: 80
targetMemory: 80
```

Run helm update to update helm deployment with new parameters

```
helm upgrade keycloak bitnami/keycloak -f values.yaml -n idmsnode
```

# 5.2 Clustering configuration

To enable Keycloak for clustering, distributed cache management has to be enabled and Node discovery set.

Configure Distributed Caching with Infinispan by enabling caching and cache stack in values.yaml:

cache: enabled: true stackFile: "" stackName: kubernetes extraEnv: keycloak: - name: KC\_CACHE value: kubernetes - name: KC\_CACHE\_STACK value: ispn

This can also be added into the keycloak-env-vars config file

kubectl edit configmap keycloak-env-vars -n idmsnode

```
KEYCLOAK_CACHE_STACK: kubernetes
KEYCLOAK_CACHE_TYPE: ispn
```

Node Discovery by Enabling JGroups

Add to values.yaml to configure node discovery. (JGROUPS\_BIND\_PORT range can be flexible)

- name: JGROUPS\_BIND\_PORT value: "17900-18000" - name: JGROUPS\_BIND\_ADDR valueFrom: fieldRef: fieldPath: status.podIP - name: JGROUPS\_STACK value: "kubernetes"

With a successful configuration, clustering activity can be verified through the logs of any instance, preferably the first one.



Figure 3: Cluster info from Infinispan in Keycloak pod logs

### 5.3 JavaScript Provider configuration

Ability to build custom scripts require the scriptsv2 feature, which is part of the preview feature. This needs to be added to the environment variable of Keycloak and initiated upon pod startup.

Edit ConfigMap

kubectl edit configmap keycloak-env-vars -n idmsnode

Add in either --features=scripts or ---features=scripts and save ConfigMap

The values.yaml file is also updated with preview and SPI enabling.

- name: KC\_SPI\_POLICY\_JS\_ENABLED
value: "true"
- name: KC\_FEATURES
value: "preview"

Prepare the JavaScript policy and Meta in a folder following the structure

```
META-INF/keycloak-scripts.json
script-example-policy.js
```

Copy the folder from local machine to providers folder in the keycloak server

```
kubectl cp ./script-example-policy \
idmsnode/keycloak-0:/opt/bitnami/keycloak/providers
```

Connect into console of the Keycloak server

kubectl exec -it keycloak-0 -n idmsnode -- bash

Compress the folder to a  $\tt.jar$  file

```
cd /opt/bitnami/keycloak/providers/
jar cvf script-example-policy.jar -C script-example-policy .
```

Initiate a Build for the .jar file

```
./opt/bitnami/keycloak/bin/kc.sh build
```

Restart Keycloak in Development mode and redirect port to avoid issues with utilized port

/opt/bitnami/keycloak/bin\$ kc.sh start-dev --http-port=8082

Perform a port forwarding to access the Keycloak over the browser

```
kubectl port-forward pod/keycloak-0 8081:8081 -n idmsnode
```

Ensure to enable "Authorization" in client settings for policy settings to be visible.

					0	admin 🔻	
Keycloak master	Clients > Create client <b>Create client</b> Clients are applications and service	is that can request auther	ntication of a user.				
Manage							
Clients	1 General settings	Client authentication	On				
Client scopes	2 Capability config	0					
Realm roles	3 Login settings	Authorization ③	On On				
Users		Authentication flow	🖌 Standard flow 💿	🗹 Direct access grants 📀			
Groups			Implicit flow 🔊	Service accounts roles 📀			
Sessions			OAuth 2.0 Device Auth	orization Grant 💿			
Events			OIDC CIBA Grant 💿				
Configure							
Realm settings							
Authentication							
Identity providers							
User federation							
		Back Next	Cancel				

Figure 4: Authorization Toggle in Client Settings

Custom Policy will be accessible in policies under clients' configuration



Figure 5: Custom Policy in Keycloak

# 6 Time Series Monitoring

#### 6.1 Prometheus

In the terminal add the Helm repository for Prometheus and update repository

```
helm repo add prometheus-community \
https://prometheus-community.github.io/helm-charts
helm repo update
```

Install the downloaded chart to accurate node and namespace

```
helm install prometheus prometheus-community/prometheus \
--namespace monitoring \
--set nodeSelector.node=monitoring
```

Get the services in namespace and expose the Prometheus-server port

```
kubectl get svc -n monitoring
kubectl expose service prometheus-server \
--type=NodePort \
--target-port=9090 \
--name=prometheus-server-ext \
-n monitoring
```

Forward port to be accessible locally and access over the **Browser**: https://127.0.0.1:9090 kubectl port-forward svc/prometheus-server-ext 9090:80 -n monitoring

< > e	J 0 127.0.0.1:9090	/targets?search=				ć	0   🦁	ລມ <b>ວ</b> ⊱
🖿 Containers 🖿 Jobs 🖿 NCI 🖿 Github 🖿 Security								
Prometheus Alerts Graph Status - F								<b>\$</b> (
Targets								
All scrape pools - All Unhealthy Collag	ose All Q Filt	er by endpoint or labels						🔽 Unknown 🔽 Unhealthy 🔽 🛛
kubernetes-apiservers (1/1 up) show less								
Endpoint	State	Labels			Last Scrape	Scrape Duration	Error	
https://52.146.154.29/metrics		Instance=152.146.154.28.44 Discovered labels:ddress_=52.146.154.28.44 Discovered labels:ddress_=52.146.154.28.44 meta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmeta_kidentes_motopmotopmetamotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopmotopm	Side Tubernite-spisorers     (     Side Tubernite-spisorers     (     Side Tubernite-spisorers     (     Side and spisorers     (     Side and spisorers     (     Side and spisorers     Side and spisorers     Side and spisorers     (     Side and spisorers     Side and	kip_mirrors*true* es_lo_kkip_mirrors*true*	20.762s ago	214.152ms		
kubernetes-nodes (5/5 up) atow kas								

Figure 6: Prometheus Portal

Live metrics can be viewed at https://<ipexposed:port>/metrics.



Figure 7: Prometheus Metrics

#### 6.2 Grafana Dashboard

In the terminal add the Helm repository for Grafana and update repository

```
helm repo add prometheus-community https://grafana.github.io/helm-charts
helm repo update
```

Install the downloaded chart to accurate node and namespace

```
helm install grafana grafana/grafana \
--namespace monitoring \
--set nodeSelector.node=monitoring
```

Confirm running pod and service in the monitoring Namespace

```
kubectl get pods -n monitoring
kubectl get svc -n monitoring
```

Retrieve the admin password for Grafana login (Username is admin)

```
kubectl get secret grafana -n monitoring \
-o jsonpath="{.data.admin-password}" | base64 --decode
```

Forward port to be accessible locally and access over the browser at **Browser**: https://127.0.0.1:3000/login

```
kubectl port-forward svc/grafana 3000:80 -n monitoring
```



Figure 8: Grafana Login page



Figure 9: Grafana Console

To ensure persistence of data settings configured in Grafana, add a Persistent Volume Claim (PVC)

```
helm upgrade grafana grafana/grafana --namespace monitoring \
--set persistence.enabled=true \
--set persistence.size=10Gi \
--set persistence.storageClassName=default
```

Confirm presence of persistent volume claim

kubectl get pvc -n monitoring

Connect Grafana to Prometheus Data Source and add dashboard of choice based on desired metrics

Ø				Q Search or	jump to		⊞ <b>#</b> +k
=	Но	me > Connections > Data sources	> prometheus				
	۵	Home 🔲	<b>t∦ Settings</b> ⊞ Dashboards				
>	Д	Bookmarks					
>	☆	Starred	Name ③ prometheus		Default 📿		
>		Dashboards					
>		Explore	Before you can use the Prometheus data source, you must configure it below or in the config file. For detailed instructions, <u>view t</u> Fields marked with * are required				
>		Alerting					
~	Ø	Connections					
	1	Add new connection					
	- [	Data sources					
>		Administration					
			Connection				
			Prometheus server URL *	onitoring svc cl	uster local		
				onitoring.ove.en	usternoour		

Figure 10: Adding Data Source

Custom query parameters	Example: max_source_resolution=5m&timeout	
HTTP method	© POST ~	
Exemplars + Add		
<ul> <li>Successfully queried to Next, you can start to</li> </ul>	t <b>he Prometheus API.</b> visualize data by <mark>building a dashboard</mark> , or by querying	data in the Explore view.
Delete Save & test		

Figure 11: Saving Data Source



Figure 12: Grafana Dashboard

# 6.3 Queries for Metrics collection

These are the PromQL queries used to grab metrics furnished to the Grafana Dashboard.

CPU usage percentage for pods in a Namespace

```
sum(rate(container_cpu_usage_seconds_total{namespace="idmsnode"}[5m])) by
(pod) * 100
```

Memory Usage in Megabytes (MB) for all pods in a Namespace

container\_memory\_usage\_bytes{namespace="idmsnode"} / 1048576

Http Requests total for pods in a Namespace

```
apiserver_request_total{resource="pods"}
```

Pod Additions into a Namespace

increase(kube\_pod\_status\_phase{namespace="idmsnode", phase="Running"}[5m])
increase(kube\_pod\_status\_phase{namespace="appnode", phase="Running"}[5m])

# 7 Load Testing

# 7.1 JMeter Configuration

Install JMeter on MAC OS.

**Note:** For other operating systems, download at <a href="https://jmeter.apache.org/download\_jmeter.cgi">https://jmeter.apache.org/download\_jmeter.cgi</a>. brew install jmeter

Verify JMeter installation

jmeter -v



Figure 13: JMeter Installed

#### Launch Jmeter

#### jmeter

Load JMeter Test Plan file from downloaded artifact at /Jmeter Files/testplan1.jmx

	• •	testplan1.jmx (/Users/ezekielayanda/Thesis/Jmeter Files/testplan1.jmx) - Apache JM	eter (5.6.3)
1	📽 🤮 📰 🗶 📭 📵 + 🗕 ·	5/ 🕨 🔊 🐵 💐 🎬 🖚 🏷 🚍 👔	00:00:00 🕂 0 0/0 😍 🦑
- 1	Test Plan  Bulk User Creation  USR223ReProcessor1  HTTP Request	Thread Group Name Bulk User Creation Comments:	
	View Hesults Irea     JSR23 PostProcessor     Bulk User Login	Action to be taken after a Sampler error      Open	Now
	CSV Data Sd Config HTTP Request Login CSV Data Sd Config HTTP Request Login View Results Tree View Results Tree	Iminuter Files       Iminuter Files       Iminuter Files       Iminuter Files       File       Testplant Jimx       File Format:       JMeter [jmx]	

Figure 14: Load Test Plan

Three Thread Groups are present for User Registration, Login and Deletion respectively.

• • • testpl	an1.jmx (/Users/ezekielayanda/Thesis/Jmeter Flles/testplan1.jmx) - Apache JMeter (5.6.3)					
🖀 🕼 🖶 🗶 📭 🗓   + - 🍫   >	🔈 💿 🗶 💐 🏙 🚓 🍾 🚍 🔞	00:00:00 🛕 0 0/0 🛟 🦑				
<ul> <li>Test Plan</li> <li>OB Bulk User Creation</li> </ul>	Thread Group					
USR223PreProcessor1  HTTP Request	Name: Bulk User Creation					
View Results Tree	Comments: Action to be taken after a Sampler error					
Gonzes Post-locesson     Gonzes Post-locesson     CSV Data Set Config	Continue Start Next Thread Loop Stop Thread Stop Test Stop Test Now     Thread Properties					
View Results Tree						
CSV Data Set Contig	Number of Threads (users): 600					
ATTP Request Login     View Results Tree	Ramp-up period (seconds): 7					
	Same user on each iteration					
1	Delay Thread creation until needed					
	Specify Thread lifetime					

Figure 15: Thread Groups

Number of users to be created can be specified with ramp up period. The user creation request is forwarded from the Java application to Keycloak where users are created almost real time. Created users data are saved in /path/userdata.csv

•••	testplan1.jmx (/Users/ezekielayanda/Thesis/Jmeter Flles/testplan1.jmx) - Apache JMeter (5.6.3)	
📑 🚳 🚔 🚍  🖬 😸 👘	≶/ ▶ № 🐵 🕸 👹 @ ≽ 🚍 👔	00:00:00 🛕 0 0/0 🛟 🦑
Lest Plan     Bulk User Creation	Thread Group	
HTTP Request	Comments: Action to be taken after a Sampler error	
Bulk User Login     CSV Data Set Config     HTTP Request Login	Continue      Start Next Thread Loop      Stop Thread      Stop Test      Stop Test      Stop Test Now	
✓ View Results Tree	Thread Properties	
CSV Data Set Config  HTTP Request Login  View Results Tree	Number of Threads (users):     2000       Ramp-up period (seconds):     7	
	Same user on each iteration  Delay Thread creation until needed	
	Specify Thread lifetime	

Figure 16: User Creation

User creation and data export to CSV are done at the JSR223 PreProcessor and Post-processor stages respectively utilizing groovy scripts saved in the artifact at:

```
/Jmeter FIles/usercreation.groovy
/Jmeter FIles/createduserexport.groovy
```



Figure 17: JSR223PreProcessor

# 7.2 (Optional) Run Login request on Azure Load testing

For more compute power, Azure load testing can be used to run the load testing due to it's more robust capacity. Load testing configuration in JMeter can be exported as as .JMX file and uploaded to Azure load testing.

You can create an Azure Load testing resource and create a new test in it by uploading the JMter script.



Figure 18: Create Test in Azure Load Testing

Upload the Jmeter file and the user data csv genrated for new users created in Keycloak

you can also choose to split the CSVs evenly between engines if you are using more than one engine (Completely optional).



Figure 19: Upload Test plan

If you have a large csv file, you can split the workload on more than one engine instance and also generate the load from different regions. As this configuration has public endpoints enabled, we have selected "Public" for the traffic mode.

Basics Test plan P	Parameters Lo	ad Monitoring	Test criteria		
Scaling your load tes 1. Set the number of engine instance. The script. We recommer maximum for your te 2. Set engine instanc reach a total of 1,000	t: threads in the JMX maximum number of d 250 threads per e est scenario. es accordingly to rea 0 threads.	file. This represents the of threads per engine i ngine instance. You ca ach the desired numbe	e number of threads (or virtue nstance varies based on the n monitor the engine health er of threads. For example, se	al users) executed by 1 complexity of your test metrics to determine the t engine instances = 4 to	
Load configuration					
Configure the number of t	est engines that w	ould be required to r	run your test. Learn more 🖻	ſ	
Engine instances * 🕕			-o		
		The current limit for engine instances" i <u>ticket</u> request to in	or "Engine instances per test r s 20 in "North Europe" regio crease limits. <u>Learn more</u> 더	un" is 10 and "Concurrent n. Please <u>create a support</u>	
Load distribution Configure load distribution test load per region	n to generate load	from multiple region	ns. Select upto 8 regions an	d the percentage of tota	al
Region	% of	load	Number of en	gines	
North Europe (default r	egion) 100		4		
+ Add/Edit Regions					
Network Configure networking to e	nable the test traff	fic to reach your publ	lic or private end points. Le		
Configure test traffic mode	e* () (	Public			
		) Private			

Figure 20: Load Configuration

Other settings can be skipped and then save configuration. Launch the test plan

Home > Azure Load Testing >						Run	
Experiment 3							
Last Updated by : ayandaezekiel@hib	pizatogs.org   Last Updated on						
\ominus View all tests   े Refresh	🕨 Run 🗧 Compare		🖇 Configure 🗸   🏢 D	elete test runs 📋 Delet	etest 📄 Save	Run test	
Nou con a superior de la state son		the second second second	·			The test will run using the following set	tings:
Tou can now schedule the tes	st to run at a specific time or a	at a recurring intervent	tor the preview now. <u>Learn me</u>	010		Test script file from Experiment 3	
<ul> <li>Test details</li> </ul>						<ul> <li>Latest configuration</li> <li>Latest app components</li> </ul>	
Name	Test ID		Test type	Descr	iption	Test run description ①	
Experiment 3	967293c	4-6444-4ba7-a614-cb6eeed	JMX	Test v	vith 5000 Users - Exclusive	Dabua modo	
						/	
Test runs Trends							If this is enabled, the test run will run in debug mode providing debug logs along with request
Search	Time range : None	Status : All					run in debug mode will run with only one
Name †	Description	Duration (in minuter)	Virtual unorr (avorag	Virtual user bours 0/	Pup by		more C
	NewPar		virtual users (averag	virtual user riours (v	Kun by		
TestRun_12/6/2024_9:14:	New Kun	4.03	310	21.57	ayandaezekiel@hibi		
TestRun_12/3/2024_12:0	Run 1 - Threshold = 5	. 5.12	947	81.61	ayandaezekiel@hibi		
TestRun_12/3/2024_8:30:	Run 2 - Threhold = 9	2.47	942	39.27	ayandaezekiel@hibi		
						<b>↓</b>	
						Run Cancel	

Figure 21: Test Plan Launch

# 7.3 (Optional) Add Backend Listener for Azure Application Insights

Backend Listeners can be added to the test plan in other to provide more logging and information about response from each requests. For setup guide, see: <a href="https://github.com/adrianmo/jmetebackend-azure">https://github.com/adrianmo/jmetebackend-azure</a>

Once added to the JMeter confugration file, you can query logs of new Azure Load testing run in real time from Application insights



Figure 22: Application Insights logs

💡 New Query 1* 🛛 🛛 🕂		🖋 Try the new Log Analytics 🛛 🛇 Feedback 🛛 🖶 Queries hub 🛛 🍪
PIDMS-App-Insight Select scope	▶ Run Time range : La	ast 48 hours $ $ 🗟 Save $\lor$ 🖄 Share $\lor$ $+$ New alert rule $\mapsto$ Export $\lor$ $\cancel{2}$ Pin to $\lor$ $ $
Tables Queries Functions ···· «	<pre>1 requests 2   project id, customM4 3 4</pre>	easurements, customDimensions
C Search :		
	Results Chart	
<sup>†=</sup> Collapse all	id	customDimensions
▲ Favorites	> 2708877742998329208	، ResponseCode":"200","TestStartTime":"1733476536196","IdleTime":"0.0","ResponseData":"{{"message\":\"Login failed,
▶ ⊞ traces	> 6386668828070260060	 {"ResponseCode":"200","TestStartTime":"1733476536264","IdleTime":"0.0","ResponseData":"(\"message\":\"Login failed
Application Insights	> 8817346858767028156	 {"ResponseCode":"200","TestStartTime":"1733476536246","IdleTime":"0.0","ResponseData":"{\"message\":\"User Access
▶	> 5665202532651261775	 {"ResponseCode":"200","TestStartTime":"1733476536323","IdleTime":"0.0","ResponseData":"{\"message\":\"No user foun
▶ 目 browserTimings	> 6004618194466348185	 {"ResponseCode":"200","TestStartTime":"1733476536196","IdleTime":"0.0","ResponseData":"{\"message\":\"User Access
El customEvents	> 6250385653109165100	 {"ResponseCode":"200","TestStartTime":"1733476536246","IdleTime":"0.0","ResponseData":"{\"message\":\"User Access
El customMetrics	> 2154559477461016658	 {"ResponseCode":"200","TestStartTime":"1733476536264","IdleTime":"0.0","ResponseData":"{{"message}":{"User Access
H dependencies	> 9032636932284130038	 {"ResponseCode":"200","TestStartTime":"1733476536323","IdleTime":"0.0","ResponseData":"{\"message\":\"No user foun
method exceptions	> 5965895296692424360	ا ResponseCode":"200","TestStartTime":"1733476536196","IdIeTime":"0.0","ResponseData":"{\"message\":\"Login failed,
▶ 目 pageViews	> 4711503575559287018	{"ResponseCode":"200","TestStartTime":"1733476536246","IdleTime":"0.0","ResponseData":"{\"message\":\"User Access
performanceCounters	> 1978715628991091001	{"ResponseCode":"200","TestStartTime":"1733476536264","IdleTime":"0.0","ResponseData":"{\"message\":\"User Access

Figure 23: App Insights Logs