



**POLYTECHNIQUE  
MONTRÉAL**

UNIVERSITÉ  
D'INGÉNIERIE

## **LOG3430 – MÉTHODES DE TEST ET DE VALIDATION DU LOGICIEL**

### **TP5 : Simulations de tests de charge**

Automne 2022

Groupe 3 (4) :

Victor Kim – 1954607

Jérémy Perreault – 1903274

Soumis à :

Maxime Lamothe

Dmytro Humenium

Adam Halim

[1 décembre 2022]

**Question 1 : Dans la partie A, combien de demandes d'URL sont envoyées? Toutes les demandes sont-elles traitées avec succès? Vous devez fournir des captures d'écran de votre plan de test, avec les captures d'écran pour les parties A.f, A.g et A.h, ainsi que les captures des résultats e.g. de View Results Tree et Summary Report dans votre rapport. Ajoutez aussi les fichiers testA.jmx et resA.jtl.**

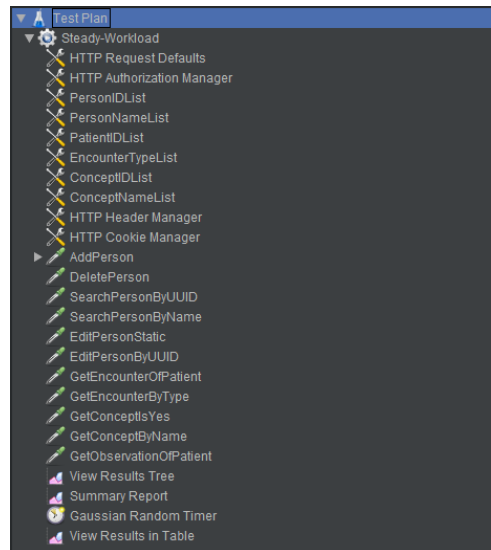


Figure 1. Plan de test

### Source de données CSV

Nom :

Commentaires :

Configuration de la source de données CSV

Nom de fichier:	<input type="text" value="data/personid.csv"/>
Encodage du fichier:	<input type="text"/>
Noms des variables (séparés par des virgules):	<input type="text" value="PERSON"/>
Ignorer la première ligne (utilisé uniquement si Noms des Variables n'est pas vide):	<input type="text" value="False"/>
Délimiteur (utiliser 't' pour la tabulation):	<input type="text" value=","/>
Autoriser les données avec des quotes ?:	<input type="text" value="False"/>
Recycler en fin de fichier (EOF) ?:	<input type="text" value="True"/>
Arrêter l'unité à la fin de fichier (EOF) ?:	<input type="text" value="False"/>
Mode de partage:	<input type="text" value="Toutes les unités"/>

Figure 2. Configuration *PersonIDList*

### Requête HTTP

Nom :

Commentaires :

▲▼

**Basique** **Avancée**

Serveur web

Protocole [http] :  Nom ou adresse IP:

Requête HTTP

Méthode :  Chemin :

☐ Rediriger automat. 
 ☒ Suivre les redirect. 
 ☒ Connexion persist. 
 ☐ Multipart/form-data 
 ☐ Entêtes compat. navigateur

Figure 3. Requête HTTP *SearchPersonBYUUID*

### Requête HTTP

Nom :

Commentaires :

▲▼

**Basique** **Avancée**

Serveur web

Protocole [http] :  Nom ou adresse IP:

Requête HTTP

Méthode :  Chemin :

☐ Rediriger automat. 
 ☒ Suivre les redirect. 
 ☒ Connexion persist. 
 ☐ Multipart/form-data 
 ☐ Entêtes compat. navigateur

**Paramètres** **Corps de la requête** **Téléchargement de fichiers**

```

1 [{"names":[{"givenName": "${__RandomString(10,abcdefghijklmnopqrstuvwxyz)}",
2   "familyName": "${__RandomString(10,abcdefghijklmnopqrstuvwxyz)}"}],
3   "gender": "P",
4   "age": "${__Random(1, 100)}"}]
  
```

Figure 4. Requête HTTP *EditPersonByUUID*

### Arbre de résultats

Nom :

Commentaires :

Écrire les résultats dans un fichier ou lire les résultats depuis un fichier CSV / JTL

Nom du fichier :

---

Rechercher :  ☐ Considérer la casse ☐ Exp. régulière

---

Texte brut	Résultat de l'échantillon
✓ AddPerson	
✓ DeletePerson	
✓ SearchPersonByUUID	
✓ SearchPersonByName	
✓ EditPersonStatic	
✓ EditPersonByUUID	
✓ GetEncounterOfPatient	
✓ GetEncounterByType	
✓ GetConceptIsYes	
✓ AddPerson	
✓ DeletePerson	
✓ GetConceptByName	
✓ GetObservationOfPatient	
✓ SearchPersonByUUID	
✓ AddPerson	
✓ DeletePerson	
✓ SearchPersonByName	
✓ SearchPersonByUUID	
✓ EditPersonStatic	
✓ SearchPersonByName	
✓ EditPersonByUUID	
✓ GetEncounterOfPatient	
✓ EditPersonStatic	
✓ GetEncounterByType	
✓ EditPersonByUUID	
✓ GetEncounterOfPatient	
✓ GetConceptIsYes	

Figure 5. View Results Tree

Rapport consolidé

Nom :

Summary Report

Commentaires :

Écrire les résultats dans un fichier ou lire les résultats depuis un fichier CSV / JTL

Nom du fichier :

result/result.jtl

Parcourir...

Uniquement :

☐ Erreurs

☒ Succès

Configurer

Libellé	# Echantillons	Moyenne	Min	Max	Ecart type	% Erreur	Débit	Ko/sec reçus	Ko/sec émis	Moy. octets
AddPerson	10	49	40	72	9.79	0.00%	34.1/min	0.62	0.19	1121.0
DeletePerson	10	40	24	84	16.57	0.00%	33.8/min	1.13	0.19	2052.0
SearchPersonByUUID	10	24	16	30	3.53	0.00%	33.5/min	0.11	0.18	206.0
SearchPersonByName	10	23	16	34	5.43	0.00%	33.6/min	0.38	0.16	694.3
EditPersonStatic	10	68	57	89	11.67	0.00%	33.5/min	0.56	0.24	1020.0
EditPersonByUUID	10	40	33	63	9.14	0.00%	33.7/min	0.56	0.24	1025.0
GetEncounterOfPatient	10	13	9	21	3.19	0.00%	33.1/min	1.76	0.18	3262.2
GetEncounterByType	10	55	5	178	67.84	0.00%	33.3/min	0.22	0.16	413.9
GetConceptsYes	10	29	25	35	3.14	0.00%	33.3/min	0.32	0.16	595.0
GetConceptByName	10	38	27	51	7.82	0.00%	33.6/min	0.42	0.17	764.5
GetObservationOfPatient	10	24	8	39	10.02	0.00%	33.4/min	4.99	0.18	9186.1
TOTAL	110	37	5	178	27.02	0.00%	5.3/sec	9.49	1.75	1849.1

Figure 6. Summary Report du Steady Workload

Comme démontré par la figure 6, soit celle démontrant le Summary Report, le nombre total de requêtes URL est de 110. Il est également possible d'observer que le pourcentage d'erreurs sur les requêtes est nul. Toutes les demandes ont donc été traitées avec succès.

**Question 2 : Pour effectuer une charge de travail aléatoire dans la partie B, nous utilisons Random Order Controller. Pouvez-vous utiliser d'autres composants pour effectuer des charges de travail aléatoires? Vous devez fournir une capture d'écran de deux possibilités dans votre rapport. Ajoutez aussi les fichiers testB.jmx et resB.jtl pour chaque charge de travail (avec les noms un peu différents).**

Les résultats obtenus en utilisant le Random Order Controller peuvent être retrouvés à la figure 7 :

Rapport consolidé

Nom :

Summary Report

Commentaires :

Écrire les résultats dans un fichier ou lire les résultats depuis un fichier CSV / JTL

Nom du fichier :

C:\Users\perre\Documents\1. ECOLE\Automne 2022\LOG3430\Labo5\Labo5resB.jtl

Parcourir...

Uniquement : ☐ Erreurs ☒ Succès

Configurer

Libellé	# Echantillons	Moyenne	Min	Max	Ecart type	% Erreur	Débit	Ko/sec reçus	Ko/sec émis	Moy. octets
AddPerson	10	43	34	56	5,81	0,00%	33,0/min	0,60	0,19	1121,0
DeletePerson	10	34	27	55	7,50	0,00%	33,6/min	1,12	0,19	2052,0
SearchPersonB...	10	20	16	23	1,99	0,00%	30,1/min	0,34	0,14	694,3
SearchPersonB...	10	19	15	26	3,89	0,00%	33,8/min	0,11	0,18	206,0
GetEncounterBy...	10	68	5	169	77,01	0,00%	31,8/min	0,21	0,16	413,9
GetObservation...	10	21	5	33	8,85	0,00%	32,5/min	4,85	0,17	9186,1
GetEncounterOf...	10	12	9	16	1,90	0,00%	34,9/min	1,85	0,19	3262,2
GetConceptByN...	10	30	26	35	2,51	0,00%	34,7/min	0,43	0,17	764,6
EditPersonStatic	10	68	59	80	5,57	0,00%	33,1/min	0,55	0,23	1020,0
GetConceptsYes	10	26	25	29	1,27	0,00%	32,9/min	0,32	0,16	595,0
EditPersonByU...	10	35	31	42	3,11	0,00%	38,4/min	0,64	0,27	1024,9
TOTAL	110	34	5	169	29,77	0,00%	5,1/sec	9,24	1,70	1849,1

Figure 7. Summary Report obtenu en utilisant le Random Order Controller

Il est également possible d'utiliser d'autres composants pour effectuer des charges de travail aléatoire telles que le Random Controller (resB2.jtl) et le Random Time Counter (resB3.jtl).

La figure 8 démontre le plan de test avec le Random Controller intégré et la figure 9 présente le résultat des tests :

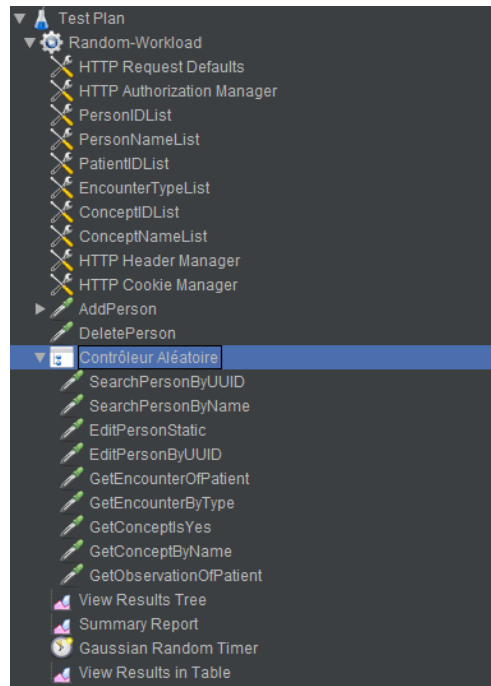


Figure 8. Intégration d'un Random Controller

**Rapport consolidé**

Nom :

Commentaires :

Écrire les résultats dans un fichier ou lire les résultats depuis un fichier CSV / JTL

Nom du fichier :   Uniquement : ☐ Erreurs ☐ Succès

Libellé	# Echantillons	Moyenne	Min	Max	Ecart type	% Erreur	Débit	Ko/sec reçus	Ko/sec émis	Moy. octets
AddPerson	10	43	36	66	8,89	0,00%	1,5/sec	1,64	0,51	1121,0
DeletePerson	10	29	22	42	5,88	0,00%	1,5/sec	3,00	0,52	2052,0
GetObservation...	2	6	6	6	0,00	0,00%	20,3/min	0,07	0,11	206,0
GetConceptByN...	2	31	29	34	2,50	0,00%	33,3/min	0,40	0,16	730,5
EditPersonStatic	2	60	59	61	1,00	0,00%	8,4/sec	8,37	3,58	1020,0
GetEncounterBy...	2	81	7	155	74,00	0,00%	1,4/sec	0,55	0,41	412,0
GetEncounterOf...	2	14	14	15	0,50	0,00%	41,2/min	2,56	0,22	3814,0
TOTAL	30	37	6	155	26,50	0,00%	4,2/sec	5,98	1,41	1469,8

Figure 9. Summary Report obtenu en utilisant le Random Controller

La figure 10 démontre le plan de test avec le Random Controller intégré et la figure 11 présente le résultat des tests :

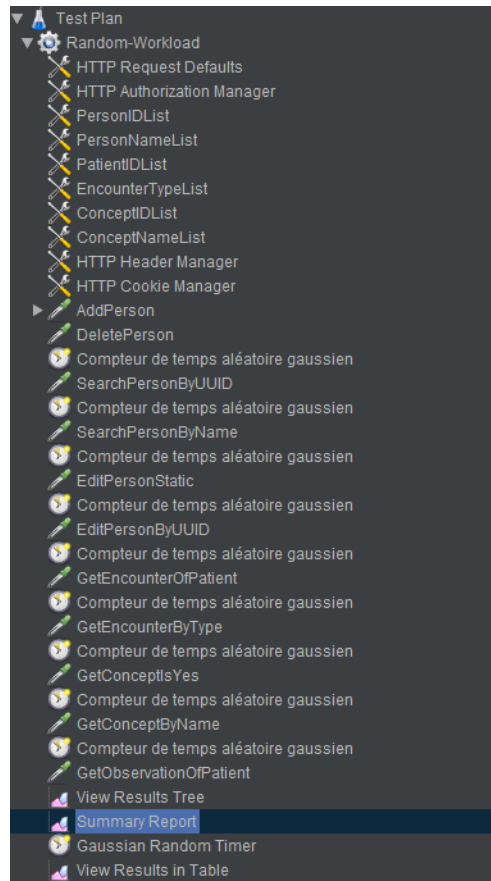


Figure 10. Intégration d'un Random Time Counter

**Rapport consolidé**

Nom :

Commentaires :

Écrire les résultats dans un fichier ou lire les résultats depuis un fichier CSV / JTL

Nom du fichier :   Uniquement : ☐ Erreurs ☒ Succès

Libellé	# Echantillons	Moyenne	Min	Max	Ecart type	% Erreur	Débit	Ko/sec reçus	Ko/sec émis	Moy. octets
AddPerson	10	43	36	68	8.99	0,00%	4,5/min	0,08	0,03	1121,0
DeletePerson	10	29	25	33	2,73	0,00%	4,5/min	0,15	0,03	2052,0
SearchPersonB...	10	19	12	28	5,16	0,00%	4,5/min	0,02	0,02	206,0
SearchPersonB...	10	17	12	24	3,74	0,00%	4,5/min	0,05	0,02	694,3
EditPersonStatic	10	67	60	76	4,25	0,00%	4,5/min	0,07	0,03	1020,0
EditPersonByU...	10	39	32	55	6,87	0,00%	4,5/min	0,08	0,03	1025,0
GetEncounterOf...	10	15	9	22	3,36	0,00%	4,5/min	0,24	0,02	3262,2
GetEncounterBy...	10	83	6	170	76,89	0,00%	4,5/min	0,03	0,02	413,9
GetConceptsYes	10	26	23	29	1,73	0,00%	4,5/min	0,04	0,02	595,0
GetConceptByN...	10	31	25	61	10,11	0,00%	4,5/min	0,06	0,02	764,6
GetObservation...	10	19	6	26	6,53	0,00%	4,5/min	0,67	0,02	9186,1
TOTAL	110	35	6	170	31,61	0,00%	40,3/min	1,21	0,22	1849,1

Figure 11. Summary Report obtenu en utilisant les Random Time Counter

**Question 3 : Fournissez votre script de plan de test dans la partie C nommée testC.jmx et votre fichier de résultat resC.jtl. Ajoutez les captures d'écran avec les tests et les résultats.**

La figure 12 présente le plan de test avec le Workload A et le Workload B et la figure 13 illustre les résultats obtenus.

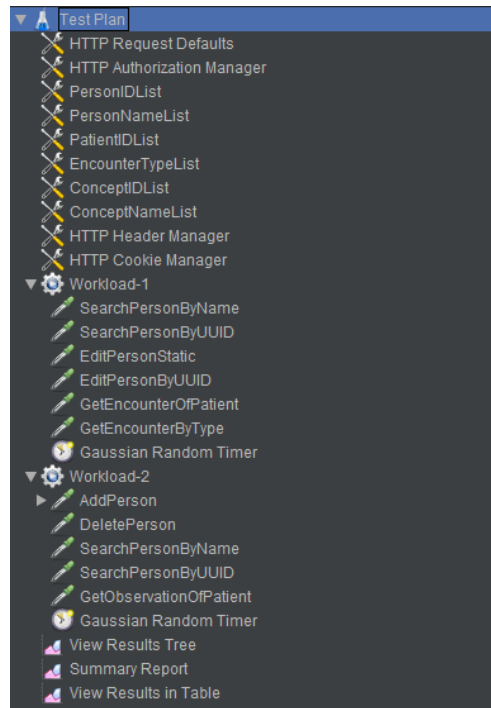


Figure 12. Plan de test incluant les Workloads A et B.

**Rapport consolidé**

Nom : Summary Report

Commentaires :

Écrire les résultats dans un fichier ou lire les résultats depuis un fichier CSV / JTL

Nom du fichier : C:\Users\pierre\Documents\1. ECOLEAutomne 2022\LOG3430\Labo5\Labo5VesC.jtl

Parcourir... Uniquement : ☐ Erreurs ☒ Succès Configurer

Libellé	# Echantillons	Moyenne	Min	Max	Ecart type	% Erreur	Débit	Ko/sec reçus	Ko/sec émis	Moy. octets
SearchPersonB...	2	82	22	143	60,50	0,00%	2,1/sec	1,37	0,55	660,0
AddPerson	1	135	135	135	0,00	0,00%	7,4/sec	8,11	2,51	1121,0
SearchPersonB...	2	18	15	22	3,50	0,00%	2,7/sec	0,53	0,83	206,0
DeletePerson	1	30	30	30	0,00	0,00%	33,3/sec	66,80	11,49	2052,0
EditPersonStatic	1	76	76	76	0,00	0,00%	13,2/sec	13,11	5,60	1020,0
EditPersonByU...	1	35	35	35	0,00	0,00%	28,6/sec	28,60	12,22	1025,0
GetObservation...	1	25	25	25	0,00	0,00%	40,0/sec	444,06	12,70	11368,0
GetEncounterOf...	1	9	9	9	0,00	0,00%	111,1/sec	22,35	35,92	206,0
GetEncounterBy...	1	171	171	171	0,00	0,00%	5,8/sec	2,35	1,72	412,0
TOTAL	11	62	9	171	56,65	0,00%	5,1/sec	8,61	1,69	1721,5

Figure 13. Résultats des Workloads A et B

**Question 4 : Collectez les données de performances du processeur (en mode utilisateur) de la partie D et fournissez un graphique d'utilisation du processeur. Ajoutez aussi les fichiers testD.jmx et resD.jtl et votre script pour créer le graphique et collecter les données de la performance (CPU, mémoire).**

La figure 14 présente le plan de test qui regroupe les quatre groupes de threads.



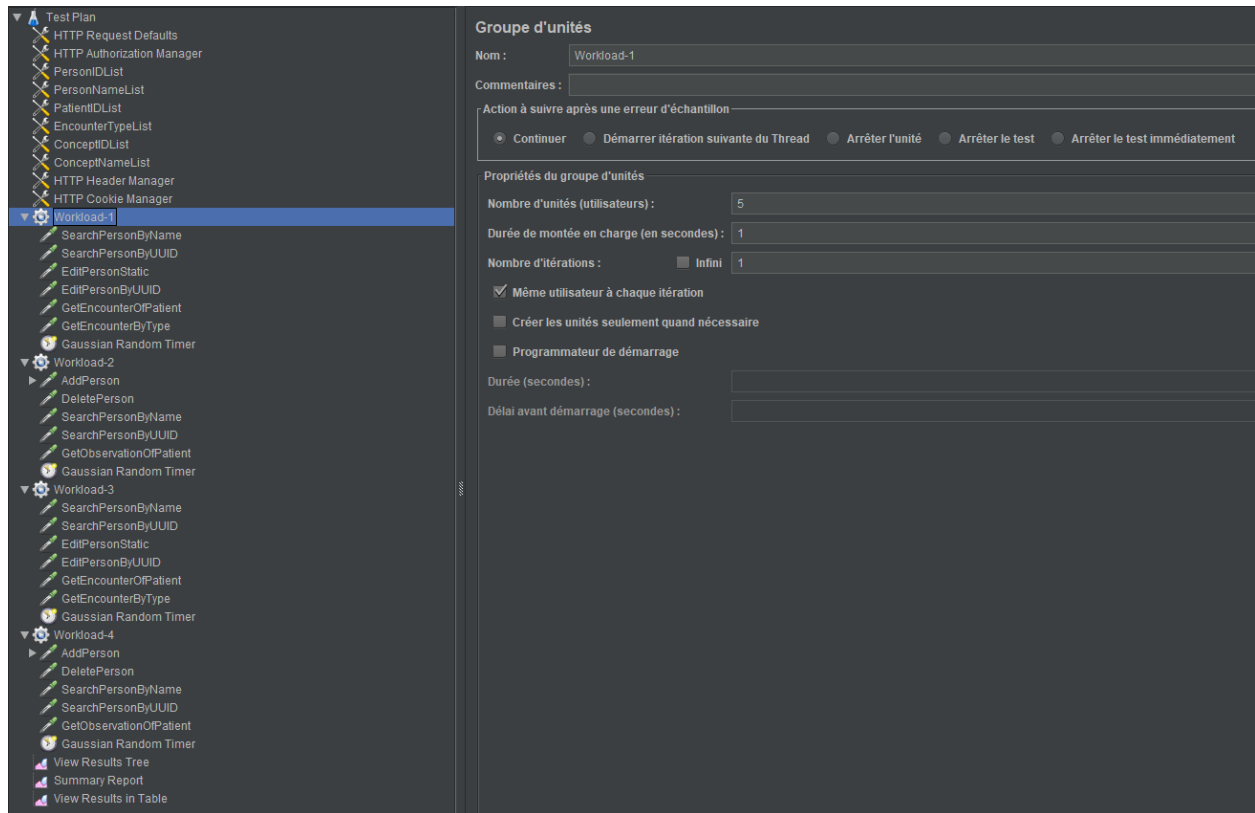


Figure 14. Plan de test incluant les quatre charges de travail

Une charge de travail variée a ensuite été lancée à l'aide du mode CLI JMeter. L'exécution de la commande est démontrée à la figure 15 :

```
PS C:\Users\perre\Documents\1. ECOLE\Automne 2022\LOG3430\Labo5\Labo5\loadTests\loadTests\apache-jmeter-5.2.1\apache-jmeter-5.2.1\bin> ./jmeter -n -t testD.jmx -l resD.jtl
Creating summariser <summary>
Created the tree successfully using testD.jmx
Starting standalone test @ Thu Dec 01 15:52:26 EST 2022 (1669927946491)
Waiting for possible Shutdown/StopTestNow/HeapDump/ThreadDump message on port 4445
summary + 32 in 00:00:03 = 9.7/s Avg: 41 Min: 7 Max: 124 Err: 0 (0.00%) Active: 5
Started: 10 Finished: 5
summary + 238 in 00:00:08 = 29.9/s Avg: 36 Min: 5 Max: 156 Err: 0 (0.00%) Active: 0
Started: 50 Finished: 50
summary = 270 in 00:00:11 = 23.9/s Avg: 36 Min: 5 Max: 156 Err: 0 (0.00%)
Tidying up ... @ Thu Dec 01 15:52:38 EST 2022 (1669927958056)
... end of run
PS C:\Users\perre\Documents\1. ECOLE\Automne 2022\LOG3430\Labo5\Labo5\loadTests\loadTests\apache-jmeter-5.2.1\apache-jmeter-5.2.1\bin> |
```

Figure 15. Lancement de la charge de travail variée

Les résultats des tests peuvent être trouvés dans le fichier resD.jtl comme démontré à la figure 16 :

```

1 1669927947283,65,SearchPersonByName,200,OK,Workload-1 1-1,text,true,,867,239,3,3,http://localh
2 1669927947283,65,SearchPersonByName,200,OK,Workload-1 1-2,text,true,,867,239,3,3,http://localh
3 1669927947567,17,SearchPersonByUUID,200,OK,Workload-1 1-2,text,true,,206,322,4,4,http://localh
4 1669927947637,17,SearchPersonByUUID,200,OK,Workload-1 1-1,text,true,,206,322,5,5,http://localh
5 1669927947647,24,SearchPersonByName,200,OK,Workload-1 1-3,text,true,,865,239,5,5,http://localh
6 1669927947745,18,SearchPersonByName,200,OK,Workload-1 1-4,text,true,,865,239,5,5,http://localh
7 1669927947849,25,SearchPersonByName,200,OK,Workload-1 1-5,text,true,,872,240,5,5,http://localh
8 1669927947813,85,EditPersonStatic,200,OK,Workload-1 1-2,text,true,,1261,436,5,5,http://localh
9 1669927947970,19,SearchPersonByUUID,200,OK,Workload-1 1-3,text,true,,206,322,5,5,http://localh
10 1669927948031,17,SearchPersonByUUID,200,OK,Workload-1 1-5,text,true,,206,322,5,5,http://localh
11 1669927948075,21,SearchPersonByUUID,200,OK,Workload-1 1-4,text,true,,206,322,5,5,http://localh
12 1669927948032,88,EditPersonStatic,200,OK,Workload-1 1-1,text,true,,1261,436,5,5,http://localh
13 1669927948287,36,EditPersonByUUID,201,Created,Workload-1 1-2,text,true,,1266,438,5,5,http://lo
14 1669927948395,90,EditPersonStatic,200,OK,Workload-1 1-5,text,true,,1261,436,5,5,http://localh
15 1669927948423,105,EditPersonStatic,200,OK,Workload-1 1-3,text,true,,1261,436,5,5,http://localh
16 1669927948539,20,GetEncounterOfPatient,200,OK,Workload-1 1-2,text,true,,4297,331,5,5,http://lo
17 1669927948481,101,EditPersonStatic,200,OK,Workload-1 1-4,text,true,,1261,436,5,5,http://localh
18 1669927948584,31,EditPersonByUUID,201,Created,Workload-1 1-1,text,true,,1266,438,5,5,http://lo
19 1669927948676,11,GetEncounterOfPatient,200,OK,Workload-1 1-1,text,true,,206,331,5,5,http://lo
20 1669927948676,34,EditPersonByUUID,201,Created,Workload-1 1-5,text,true,,1266,438,5,5,http://lo
21 1669927948781,88,GetEncounterByType ,200,OK,Workload-1 1-2,text,true,,500,312,5,5,http://local
22 1669927948745,124,GetEncounterByType ,200,OK,Workload-1 1-1,text,true,,492,302,5,5,http://loca
23 1669927948909,33,EditPersonByUUID,201,Created,Workload-1 1-3,text,true,,1266,438,3,3,http://lo
24 1669927948962,34,EditPersonByUUID,201,Created,Workload-1 1-4,text,true,,1266,438,3,3,http://lo
25 1669927949030,16,GetEncounterOfPatient,200,OK,Workload-1 1-5,text,true,,6635,331,3,3,http://lo
26 1669927949273,15,GetEncounterOfPatient,200,OK,Workload-1 1-3,text,true,,6051,331,3,3,http://lo
27 1669927949371,18,GetEncounterOfPatient,200,OK,Workload-1 1-4,text,true,,3713,331,3,3,http://lo
28 1669927949480,7,GetEncounterByType ,200,OK,Workload-1 1-3,text,true,,491,303,3,3,http://localh
29 1669927949544,7,GetEncounterByType ,200,OK,Workload-1 1-5,text,true,,492,302,2,2,http://localh
30 1669927949551,7,GetEncounterByType ,200,OK,Workload-1 1-1,text,true,,492,301,1,1,http://localh

```

Figure 16. Résultats des quatre charges de travail

L'ID du processus openmrs-standalone.jar a ensuite été identifié grâce à la commande : `jps -l | findstr "org.openmrs.standalone.ApplicationController"`. Le résultat obtenu, soit 12256, est affiché à la figure 17 :

```

PS C:\Users\perre\Documents\1. ECOLE\Automne 2022\LOG3430\Labo5\Labo5\loadTests\loadTests\OpenMRS-2.9.
0\OpenMRS-2.9.0> jps -l | findstr "org.openmrs.standalone.ApplicationController"
12256 org.openmrs.standalone.ApplicationController

```

Figure 17. ID du processus openmrs-standalone.jar