



**Cairo University**  
**Faculty of Computers and Artificial Intelligence**

---



**SLB423-Selected Labs in Software Engineering: Project**  
**Phase - 4**

**Delivered To: TA.Khaled Ibrahim**

**Prepared By:**

**Mootaz Medhat Ezzat Abdelwahab (20206074)**

**Salma Hossam elden Hassan (20206031)**

Academic Year: 2023/2024

First Semester

SLB423-Selected Labs in Software Engineering

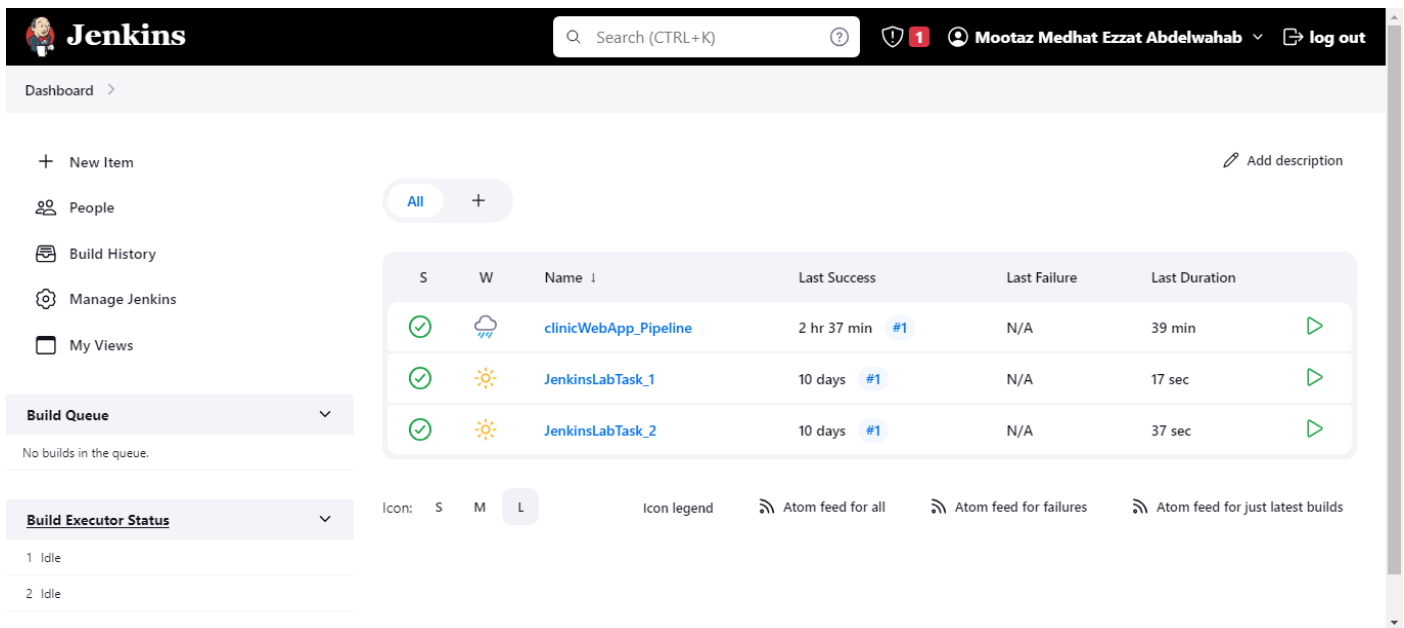
[Project - Phase 4]

---

➤ CONTENT:

- Creating pipeline for backend and frontend layer using jenkins.
- Doing performance testing using Jmeter and creating report for it after deploying on openshift.

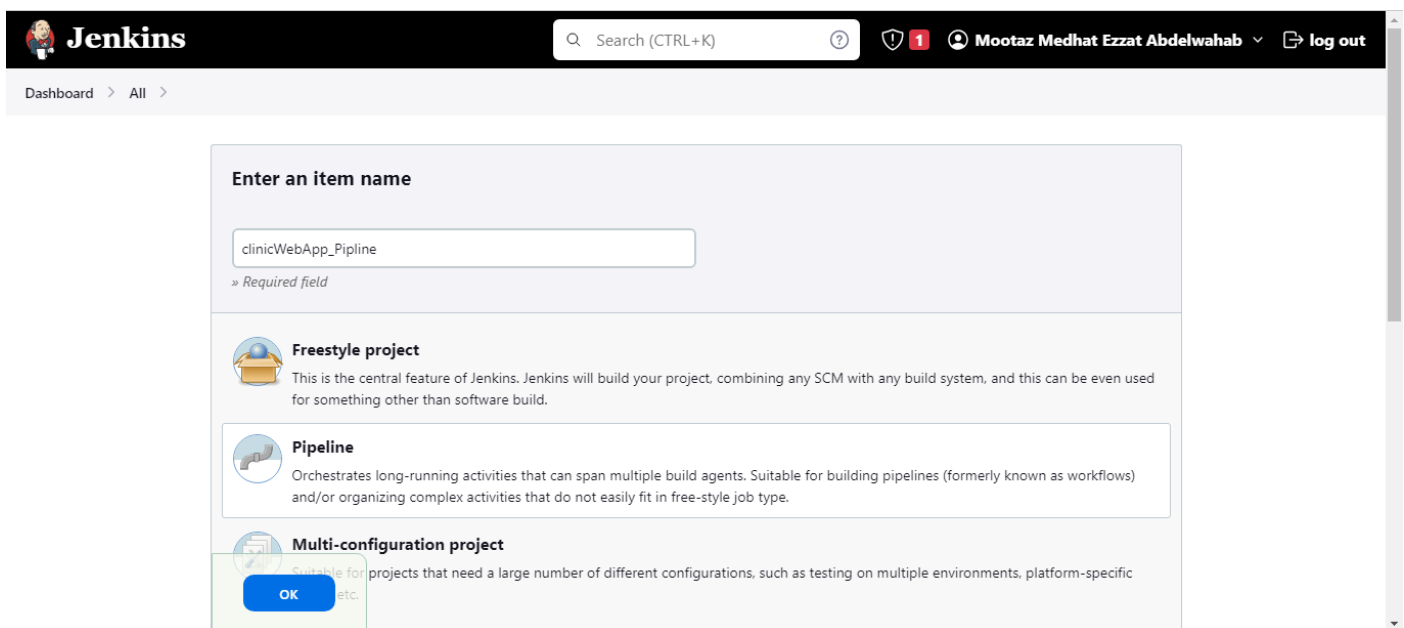
## ➤ Creating pipeline for backend and frontend layer using jenkins



The screenshot shows the Jenkins Dashboard. The top navigation bar includes the Jenkins logo, a search bar, and a user profile for 'Mootaz Medhat Ezzat Abdelwahab' with a 'log out' button. The left sidebar contains links to 'New Item', 'People', 'Build History', 'Manage Jenkins', and 'My Views'. The main content area displays a table of build items. The table has columns for 'S' (Success), 'W' (Warning), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. Three items are listed: 'clinicWebApp\_Pipeline' (2 hr 37 min, 39 min), 'JenkinsLabTask\_1' (10 days, 17 sec), and 'JenkinsLabTask\_2' (10 days, 37 sec). Below the table, there is a 'Build Queue' section showing 'No builds in the queue.' and a 'Build Executor Status' section showing two idle executors. The bottom of the dashboard includes an 'Icon legend' and three Atom feed links.

S	W	Name	Last Success	Last Failure	Last Duration
✓	☁	clinicWebApp_Pipeline	2 hr 37 min #1	N/A	39 min
✓	☀	JenkinsLabTask_1	10 days #1	N/A	17 sec
✓	☀	JenkinsLabTask_2	10 days #1	N/A	37 sec

## • Create a new pipeline item on jenkins



The screenshot shows the 'Enter an item name' dialog in Jenkins. The dialog has a text input field containing 'clinicWebApp\_Pipeline'. Below the input field, there is a 'Required field' label. The dialog also displays three options: 'Freestyle project', 'Pipeline', and 'Multi-configuration project'. The 'Pipeline' option is selected. The 'Multi-configuration project' option is partially visible at the bottom. An 'OK' button is located at the bottom right of the dialog.

**Enter an item name**

clinicWebApp\_Pipeline

» Required field

**Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other than software build.

**Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.

**Multi-configuration project**  
Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific etc.

OK

In the Pipeline section, you can either write a pipeline script directly or link to a repository. Since i have pushed the script to a repository, i will choose "Pipeline script from SCM" and select the version control system (in our case, Git).

- Set the Repository URL.

The screenshot shows the Jenkins 'Configure' page for a pipeline named 'clinicWebApp\_Pipeline'. The 'Pipeline' tab is selected in the left sidebar. Under the 'Pipeline' section, the 'Definition' is set to 'Pipeline script from SCM'. The 'SCM' is set to 'Git'. A 'Repositories' section is expanded, showing a 'Repository URL' of 'https://github.com/MootazMedhatEzzat/SLB423-Clinic-Appointment-Reservations-WebApp' and 'Credentials' set to '- none -'. 'Save' and 'Apply' buttons are at the bottom.

Specify the branch (in our case, Phase-4).

- Set the Script Path to Jenkinsfile.

The screenshot shows the same Jenkins 'Configure' page, but now the 'Branches to build' section is expanded. The 'Branch Specifier (blank for \'any\')' is set to '\*/Phase-4'. Below this, the 'Repository browser' is set to '(Auto)'. The 'Script Path' is set to 'Jenkinsfile'. 'Save' and 'Apply' buttons are at the bottom.

• Create a CI/CD for this by configuring jenkins to pull the repo and execute the bash In your Git repository, create a file named Jenkinsfile that will include the pipeline script.

Now, Jenkins will automatically pull the repository, execute the Bash script, and display the results in the console output.

Jenkins

Search (CTRL+K)

1

Mootaz Medhat Ezzat Abdelwahab

log out

Dashboard > clinicWebApp\_Pipeline >

Status

Changes

Build Now

Configure

Delete Pipeline

Full Stage View

Rename

Pipeline Syntax

Build History

trend

Filter builds...

clinicWebApp\_Pipeline

Add description

Disable Project

Stage View

Average stage times:  
(Average full run time: ~39min 3s)

	Declarative: Checkout SCM	Build and Deploy Database	Build and Deploy Backend	Build and Deploy Frontend
#4 Dec 31 13:43 No Changes	1min 40s	10min 34s	5min 22s	18min 23s

Jenkins

Search (CTRL+K)

1

Mootaz Medhat Ezzat Abdelwahab

log out

Dashboard > clinicWebApp\_Pipeline > #1

Status

Changes

Console Output

View as plain text

Edit Build Information

Delete build '#1'

Git Build Data

Restart from Stage

Replay

Pipeline Steps

Workspaces

Console Output

Started by user Mootaz Medhat Ezzat Abdelwahab  
Obtained Jenkinsfile from git <https://github.com/MootazMedhatEzzat/SLB423-Clinic-Appointment-Reservations-WebApp>  
[Pipeline] Start of Pipeline  
[Pipeline] node  
Running on Jenkins in C:\ProgramData\Jenkins\jenkins\workspace\clinicWebApp\_Pipeline  
[Pipeline] {  
[Pipeline] stage  
[Pipeline] { (Declarative: Checkout SCM)  
[Pipeline] checkout  
The recommended git tool is: git.exe  
No credentials specified  
> git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\jenkins\workspace\clinicWebApp\_Pipeline\.git # timeout=10  
Fetching changes from the remote Git repository  
> git.exe config remote.origin.url <https://github.com/MootazMedhatEzzat/SLB423-Clinic-Appointment-Reservations-WebApp> # timeout=10  
Fetching upstream changes from <https://github.com/MootazMedhatEzzat/SLB423-Clinic-Appointment-Reservations-WebApp>  
> git.exe --version # timeout=10  
> git --version # 'git version 2.42.0.windows.2'  
  
> git.exe fetch --tags --force --progress -- <https://github.com/MootazMedhatEzzat/SLB423-Clinic-Appointment-Reservations-WebApp> +refs/heads/\*:refs/remotes/origin/\* # timeout=10  
> git.exe rev-parse "refs/remotes/origin/Phase-4^{commit}" # timeout=10  
Checking out Revision a9c9a81127cadb2ec1a0b1a4f1c9bf7e8ac7fb0c (refs/remotes/origin/Phase-4)  
> git.exe config core.sparsecheckout # timeout=10  
> git.exe checkout -f a9c9a81127cadb2ec1a0b1a4f1c9bf7e8ac7fb0c # timeout=10  
Commit message: "Update README.md"  
First time build. Skipping changelog.  
[Pipeline] }  
[Pipeline] // stage  
[Pipeline] withEnv  
[Pipeline] {  
[Pipeline] stage  
[Pipeline] { (Build and Deploy Database)  
[Pipeline] script  
[Pipeline] {  
[Pipeline] powershell  
powershell.exe : #0 building with "default" instance using docker driver  
At C:\ProgramData\Jenkins\jenkins\workspace\clinicWebApp\_Pipeline@tmp\durable-0bb46d1d\powershellWrapper.ps1:3 char:1  
& powershell -NoProfile -NonInteractive -ExecutionPolicy Bypass -Comm ...  
+ ~~~~~  
+ CategoryInfo : NotSpecified: (#0 building wit...g docker driver:String) [], RemoteException  
+ FullyQualifiedErrorId : NativeCommandError  
~~~~~  
[Pipeline] }  
[Pipeline] }  
[Pipeline] }  
[Pipeline] }

```
#1 [internal] load .dockerignore
#1 DONE 0.9s

#2 [internal] load build definition from Dockerfile
#2 DONE 0.9s

#2 [internal] load build definition from Dockerfile
#2 transferring dockerfile:
#2 transferring dockerfile: 950B 2.6s done
#2 ...

#1 [internal] load .dockerignore
#1 transferring context: 2B 1.6s done
#1 DONE 7.1s

#2 [internal] load build definition from Dockerfile
#2 DONE 9.6s

#3 [internal] load metadata for docker.io/library/postgres:16-alpine3.18
#3 DONE 39.3s

#4 [internal] load build context
#4 DONE 0.0s
```

```
#5 [1/3] FROM docker.io/library/postgres:16-alpine3.18@sha256:4561d264394acf3374588df017c33cc3c7d88f68d6fdc5022efa54167ce20638
#5 DONE 1.0s

#4 [internal] load build context
#4 transferring context: 28B 0.0s
#4 transferring context: 30B 0.2s done
#4 DONE 2.4s

#6 [2/3] COPY init.sql /docker-entrypoint-initdb.d/
#6 CACHED

#7 [3/3] RUN mkdir -p /var/lib/postgresql/data/custom-data && chown -R postgres:postgres /var/lib/postgresql/data/custom-data
#7 CACHED

#8 exporting to image
#8 exporting layers
#8 exporting layers 0.1s done
#8 writing image sha256:b5e1930cc1f0ba4625d7cf7f3e5581a90776b85ed74ed47983b801b85ffd3bf5
#8 writing image sha256:b5e1930cc1f0ba4625d7cf7f3e5581a90776b85ed74ed47983b801b85ffd3bf5 0.8s done
#8 naming to docker.io/library/clinic-web-database-image
#8 naming to docker.io/library/clinic-web-database-image 2.2s done
#8 DONE 3.1s

What's Next?
```

```
View a summary of image vulnerabilities and recommendations â†’ docker scout quickview
075fc58eb50b60253161ae1f8bd5aa07012639334c8f8843164023330a7fb8
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build and Deploy Backend)
[Pipeline] script
[Pipeline] {
[Pipeline] powershell
powershell.exe : #0 building with "default" instance using docker driver
At C:\ProgramData\Jenkins\jenkins\workspace\clinicWebApp_Pipeline@tmp\durable-d5a04583\powershellWrapper.ps1:3 char:1
+ & powershell -NoProfile -NonInteractive -ExecutionPolicy Bypass -Comm ...
+ ~~~~~
+ CategoryInfo          : NotSpecified: (#0 building wit...g docker driver:String) [], RemoteException
+ FullyQualifiedErrorId : NativeCommandError

#1 [internal] load .dockerignore
#1 transferring context:
#1 transferring context: 2B 1.5s done
#1 ...
```

```
#2 [internal] load build definition from Dockerfile
#2 transferring dockerfile: 764B 2.8s done
#2 DONE 6.0s

#1 [internal] load .dockerignore
#1 DONE 7.0s

#3 [internal] load metadata for docker.io/library/node:20.9.0-alpine3.18
#3 DONE 12.4s

#4 [1/7] FROM docker.io/library/node:20.9.0-alpine3.18@sha256:cb2301e2c5fe3165ba2616591efe53b4b6223849ac0871c138f56d5f7ae8be4b
#4 DONE 0.0s

#5 [internal] load build context
#5 transferring context:
#5 transferring context: 1.04kB 6.3s
#5 transferring context: 79.69kB 8.2s done
#5 DONE 9.8s

#6 [3/7] COPY package*.json ./
#6 CACHED

#7 [4/7] RUN npm install
#7 CACHED
```

```
#8 [5/7] COPY src ./src
#8 CACHED

#9 [6/7] COPY database.js ./
#9 CACHED

#10 [2/7] WORKDIR /clinicWebApp/server
#10 CACHED

#11 [7/7] COPY server.js ./
#11 CACHED

#12 exporting to image
#12 exporting layers done
#12 writing image sha256:4e2de9c35ec26b08ecc4032f7a867b25a37bfec0fa568b42578a4d411a4f03b9
#12 writing image sha256:4e2de9c35ec26b08ecc4032f7a867b25a37bfec0fa568b42578a4d411a4f03b9 0.5s done
#12 naming to docker.io/library/clinic-web-server
#12 naming to docker.io/library/clinic-web-server 3.8s done
#12 DONE 4.3s
```

```
What's Next?

View a summary of image vulnerabilities and recommendations â€” docker scout quickview
1239dbae78310d6ff89c763c5c1e561b774159f038d6ff3681a15307243a189a

[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (Build and Deploy Frontend)
[Pipeline] script
[Pipeline] {
[Pipeline] powershell
powershell.exe : #0 building with "default" instance using docker driver
At C:\ProgramData\Jenkins\jenkins\workspace\clinicWebApp_Pipeline@tmp\durable-fcc3ecal\powershellWrapper.ps1:3 char:1
+ & powershell -NoProfile -NonInteractive -ExecutionPolicy Bypass -Comm ...
+ ~~~~~
+ CategoryInfo          : NotSpecified: (#0 building wit...g docker driver:String) [], RemoteException
+ FullyQualifiedErrorId : NativeCommandError

#1 [internal] load build definition from Dockerfile
#1 transferring dockerfile:
#1 transferring dockerfile: 955B 4.8s done
#1 ...
```

```
#2 [internal] load .dockerignore
#2 transferring context: 2B 2.5s done
#2 DONE 9.5s

#1 [internal] load build definition from Dockerfile
#1 DONE 12.5s

#3 [internal] load metadata for docker.io/library/node:20.9.0-alpine3.18
#3 DONE 26.9s

#4 [1/8] FROM docker.io/library/node:20.9.0-alpine3.18@sha256:cb2301e2c5fe3165ba2616591efe53b4b6223849ac0871c138f56d5f7ae8be4b
#4 DONE 0.0s

#5 [internal] load build context
#5 transferring context:
#5 transferring context: 38B 551.7s
#5 transferring context: 750.42kB 553.3s done
#5 DONE 556.2s

#6 [4/8] RUN npm install
#6 CACHED

#7 [5/8] COPY src ./src
#7 CACHED
```

```
#8 [6/8] COPY public ./public
#8 CACHED

#9 [7/8] COPY server.js ./
#9 CACHED

#10 [2/8] WORKDIR /clinicWebApp/client
#10 CACHED

#11 [3/8] COPY package*.json ./
#11 CACHED

#12 [8/8] RUN npm run build
#12 CACHED

#13 exporting to image
#13 exporting layers done
#13 writing image sha256:1dab3ad23d226a023ddb860a3400039e8c5808674a6f19ceb2be5dad0abc9462
#13 writing image sha256:1dab3ad23d226a023ddb860a3400039e8c5808674a6f19ceb2be5dad0abc9462 1.0s done
#13 naming to docker.io/library/clinic-web-client
#13 naming to docker.io/library/clinic-web-client 1.0s done
#13 DONE 2.0s
```

```
What's Next?
  View a summary of image vulnerabilities and recommendations at 'docker scout quickview
a6829569a80fc5d211af3db63a3f8adc7db793011b4ce9170c3e4f41da6f1
[Pipeline] }
[Pipeline] // script
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS
```

```
Command Prompt
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

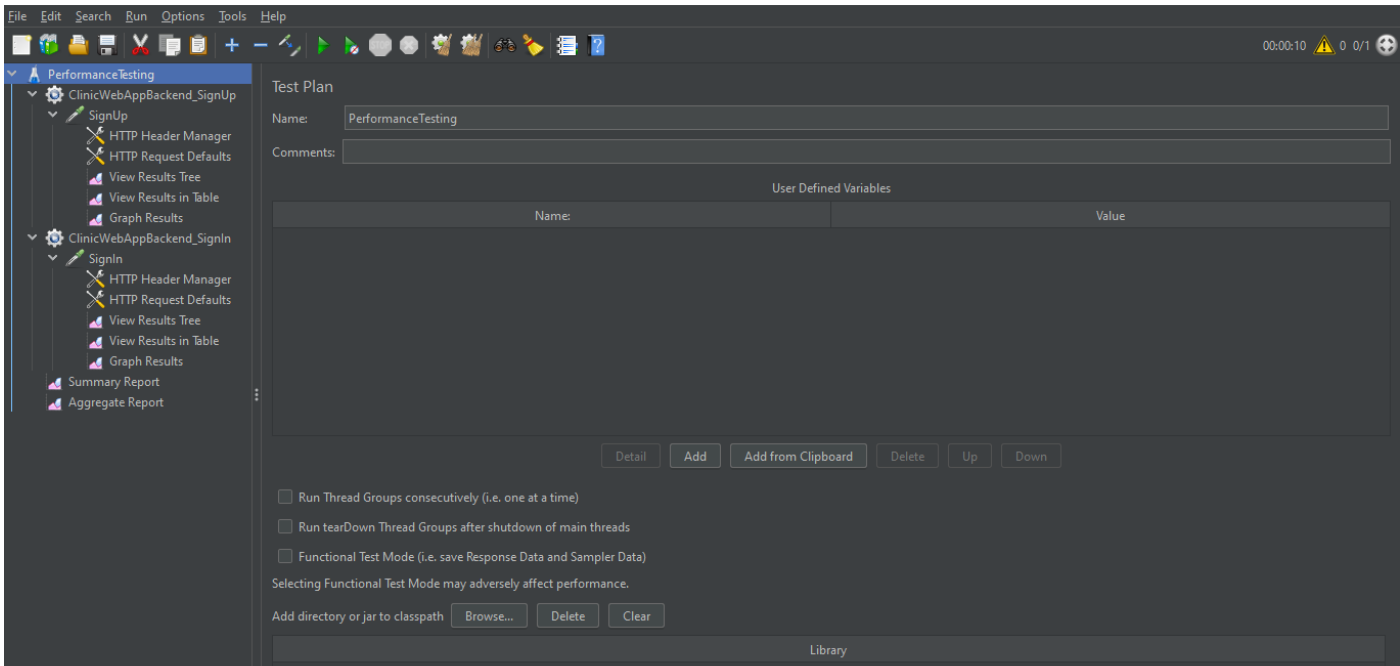
C:\Users\MAS>docker network ls
NETWORK ID          NAME                DRIVER              SCOPE
b049acc24f81        api-network         bridge              local
d5269ce54eb2        bridge              bridge              local
9fc8e2bc35d0        host                host                local
6173b4465e6d        none                null                local
bc20406df762        ui-network          bridge              local

C:\Users\MAS>docker images
REPOSITORY           TAG                IMAGE ID            CREATED             SIZE
clinic-web-client     latest             1dab3ad23d22        36 hours ago       422MB
clinic-web-database-image latest             b5e1930cc1f0        41 hours ago       239MB
clinic-web-server     latest             4e2de9c35ec2        41 hours ago       146MB

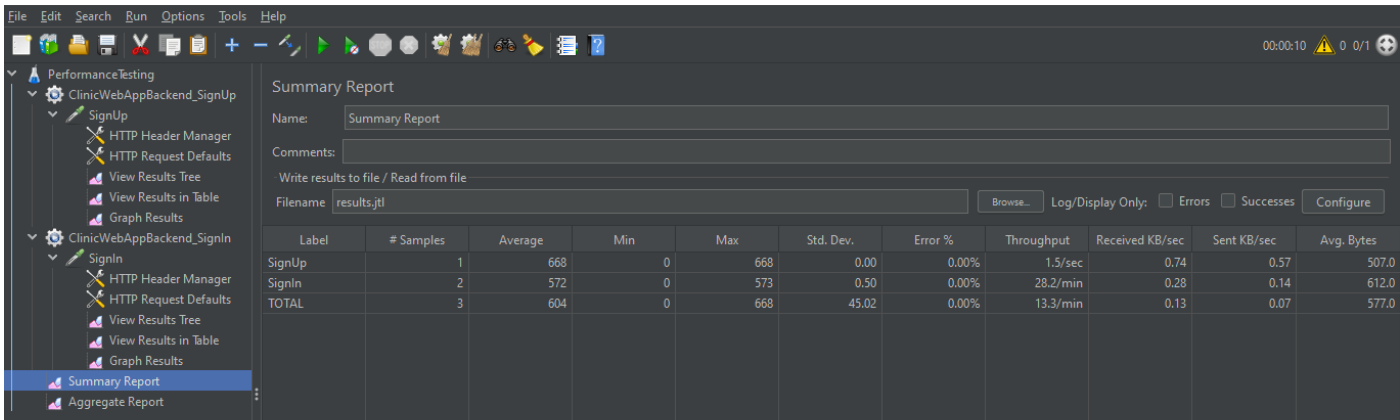
C:\Users\MAS>docker ps
CONTAINER ID        IMAGE               COMMAND              CREATED             STATUS              PORTS              NAMES
a6829569a80f        clinic-web-client:latest "docker-entrypoint.s..." 48 minutes ago     Up 48 minutes      0.0.0.0:3001->3001/tcp clinic-web-frontend
1239dbae7831        clinic-web-server:latest "docker-entrypoint.s..." About an hour ago   Up About an hour   0.0.0.0:3000->3000/tcp clinic-web-backend
075fc58eb5         clinic-web-database-image "docker-entrypoint.s..." About an hour ago   Up 28 minutes      0.0.0.0:5432->5432/tcp clinic-web-database-container
```



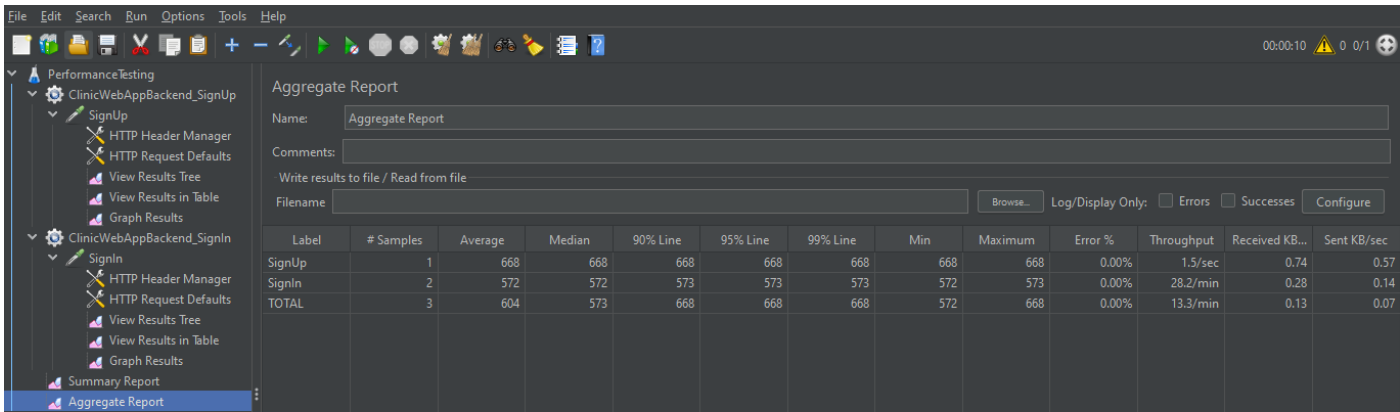
➤ Doing performance testing using Jmeter and creating report for it after deploying on openshift



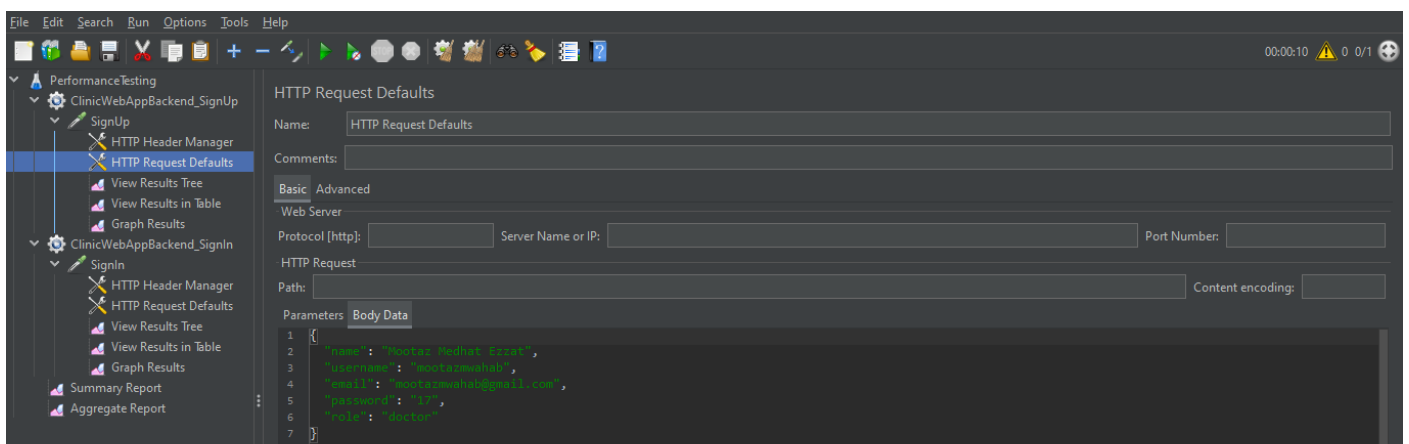
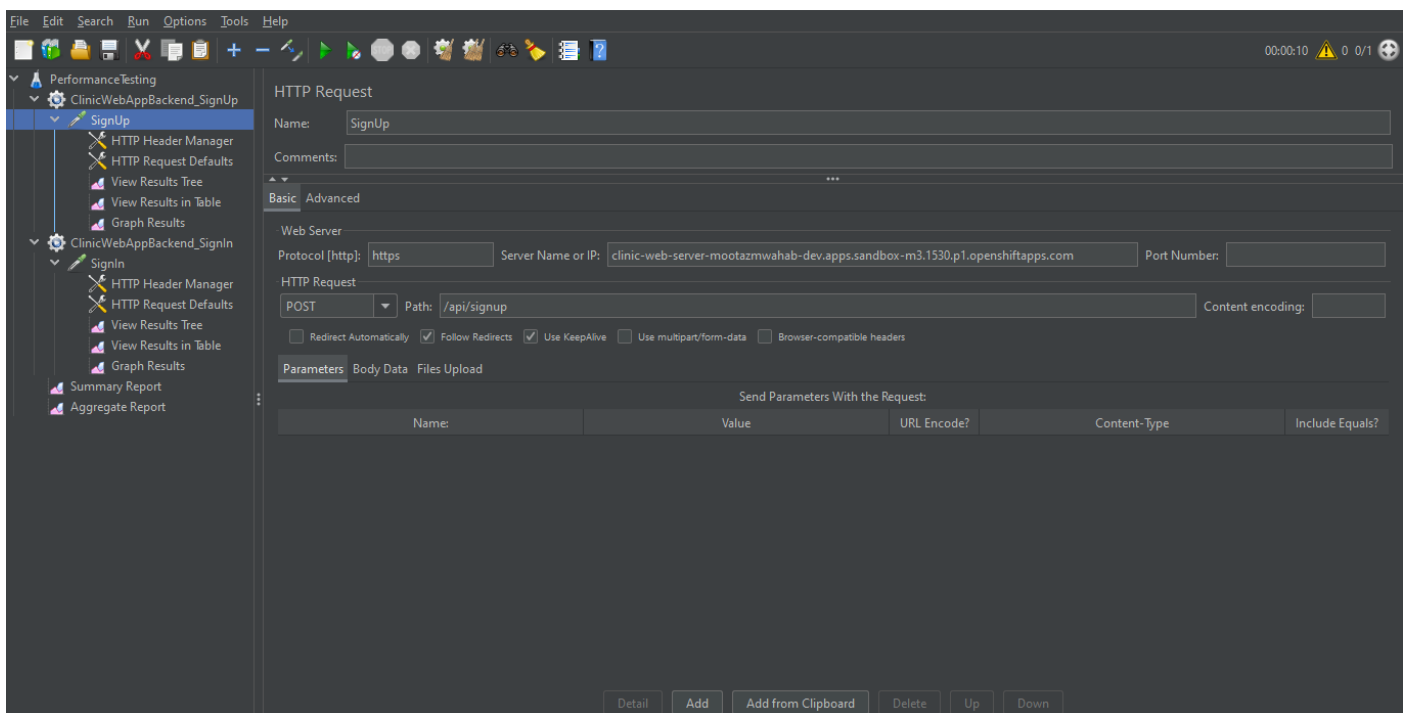
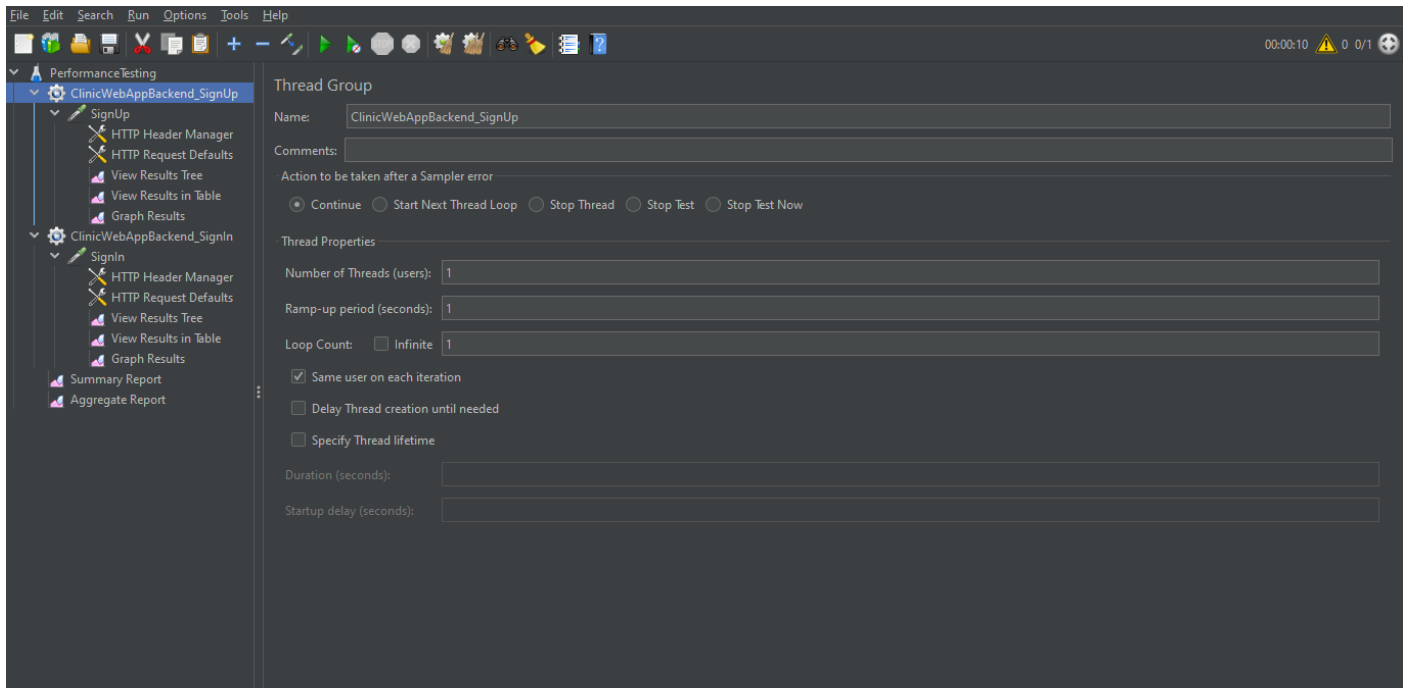
• Summary Report



• Aggregate Report



## • ClinicWebAppBackend\_SignUp



## • ClinicWebAppBackend\_SignIn

This screenshot shows the configuration for a Thread Group in Apache JMeter. The left sidebar displays a project tree with 'PerformanceTesting' expanded, showing 'ClinicWebAppBackend\_SignUp' and 'ClinicWebAppBackend\_SignIn'. The 'ClinicWebAppBackend\_SignIn' thread group is selected. The main panel shows the 'Thread Group' configuration with the following details:

- Name:** ClinicWebAppBackend\_SignIn
- Comments:** (empty)
- Action to be taken after a Sampler error:** ☒ Continue, ☐ Start Next Thread Loop, ☐ Stop Thread, ☐ Stop Test, ☐ Stop Test Now
- Thread Properties:**
  - Number of Threads (users):** 1
  - Ramp-up period (seconds):** 1
  - Loop Count:** ☐ Infinite, 1
  - ☒ Same user on each iteration
  - ☐ Delay Thread creation until needed
  - ☐ Specify Thread lifetime
  - Duration (seconds):** (empty)
  - Startup delay (seconds):** (empty)

This screenshot shows the configuration for an HTTP Request sampler within the 'ClinicWebAppBackend\_SignIn' thread group. The 'Signin' sampler is selected. The main panel shows the 'HTTP Request' configuration with the following details:

- Name:** Signin
- Comments:** (empty)
- Basic Tab:**
  - Web Server:**
    - Protocol [http]:** https
    - Server Name or IP:** clinic-web-server-mootazmwahab-dev.apps.sandbox-m3.1530.p1.openshiftapps.com
    - Port Number:** (empty)
  - HTTP Request:**
    - Method:** POST
    - Path:** /api/signin
    - Content encoding:** (empty)
    - ☐ Redirect Automatically, ☒ Follow Redirects, ☒ Use KeepAlive, ☐ Use multipart/form-data, ☐ Browser-compatible headers
- Parameters Tab:**
  - Send Parameters With the Request:** (empty table)

This screenshot shows the configuration for an HTTP Request Defaults sampler within the 'ClinicWebAppBackend\_SignIn' thread group. The 'HTTP Request Defaults' sampler is selected. The main panel shows the 'HTTP Request Defaults' configuration with the following details:

- Name:** HTTP Request Defaults
- Comments:** (empty)
- Basic Tab:**
  - Web Server:**
    - Protocol [http]:** (empty)
    - Server Name or IP:** (empty)
    - Port Number:** (empty)
  - HTTP Request:**
    - Path:** (empty)
    - Content encoding:** (empty)
- Parameters Tab:**
  - Body Data:**

```
1 {
2   "username": "mootazmwahab",
3   "password": "123"
4 }
```