

## TP 5 : Mise en place de la stack ELK : Elasticsearch, Kibana et Logstash avec Docker Compose

La gestion et l'analyse des logs sont essentielles pour surveiller et optimiser le fonctionnement d'une application. La stack ELK (Elasticsearch, Logstash et Kibana) est une solution open-source permettant de **collecter, stocker et visualiser** les logs de manière centralisée.

- **Elasticsearch** est un moteur de recherche et d'indexation qui stocke les logs et permet de les interroger rapidement.
- **Logstash** collecte, transforme et envoie les logs vers Elasticsearch.
- **Kibana** offre une interface graphique pour visualiser et analyser les logs.

Ce document décrit **pas à pas** la mise en place de la stack ELK en utilisant **Docker Compose** pour simplifier le déploiement. Nous verrons également **comment configurer une application Stock-MS** afin qu'elle génère des logs exploitables par ELK.

### 1. Installation des outils Elasticsearch, Kibana et Logstash

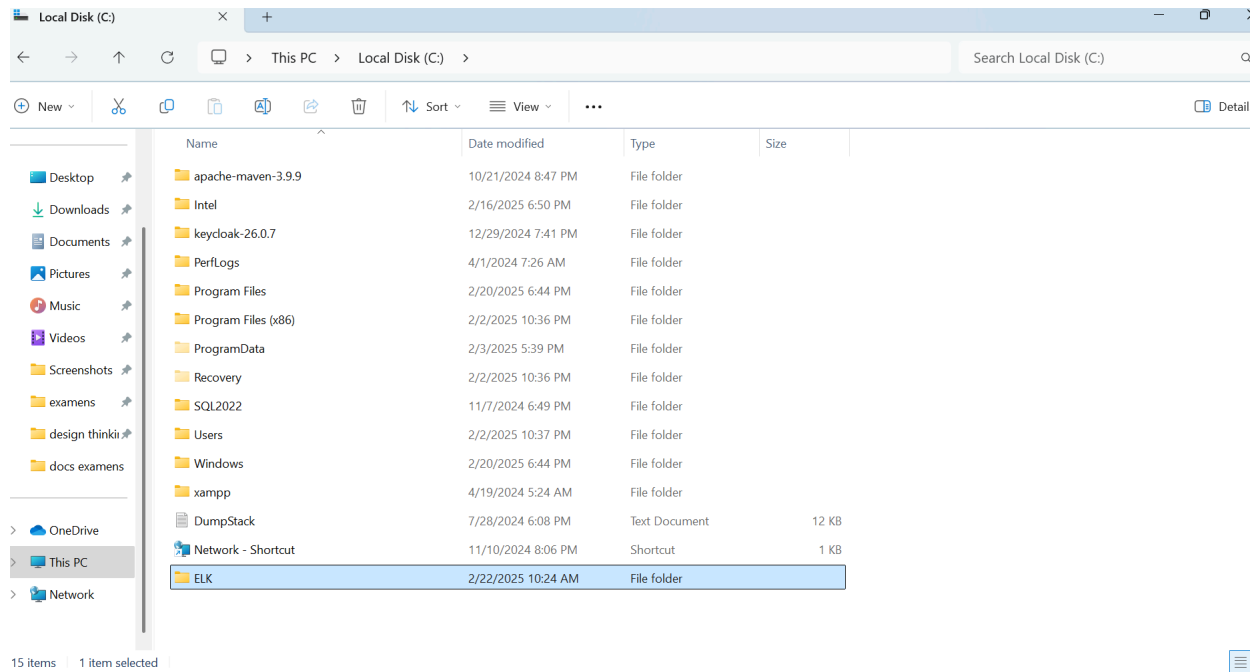
Avant de configurer la stack ELK, il est nécessaire d'installer les composants individuellement pour s'assurer de leur bon fonctionnement.

#### 1.1 Téléchargement et installation

Les fichiers d'installation des trois services peuvent être téléchargés depuis le site officiel d'Elastic :

- **Elasticsearch** : <https://www.elastic.co/downloads/elasticsearch>
- **Kibana** : <https://www.elastic.co/downloads/kibana>
- **Logstash** : <https://www.elastic.co/downloads/logstash>

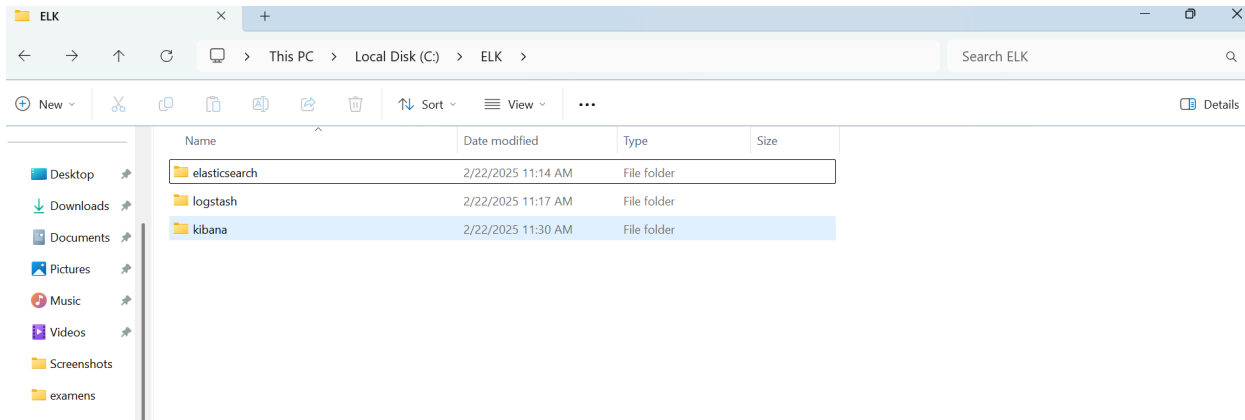
Après le téléchargement, extraire chaque fichier ZIP dans un répertoire ELK.



Exemple pour Elasticsearch :

```
unzip elasticsearch-8.17.2.zip -d /path/to/install
```

Faire la même chose pour Kibana et Logstash.



## 2. Configuration des services

### 2.1 Configuration d'Elasticsearch

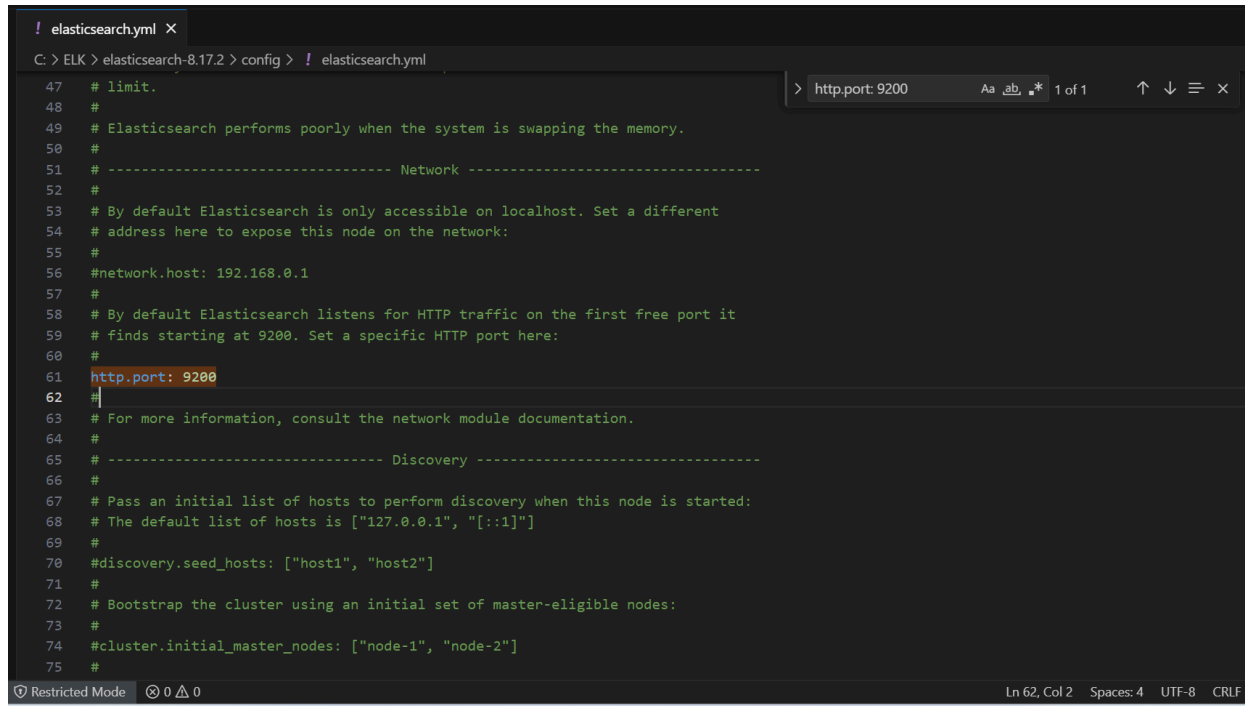
Elasticsearch est le moteur de recherche et d'indexation qui stocke les logs de manière efficace.

#### 2.1.1 Configuration du fichier `elasticsearch.yml`

Ouvrir le fichier de configuration situé dans `config/elasticsearch.yml` et modifier les paramètres suivants :

`http.port: 9200`

`xpack.security.enabled: true`



```
! elasticsearch.yml X
C: > ELK > elasticsearch-8.17.2 > config > ! elasticsearch.yml

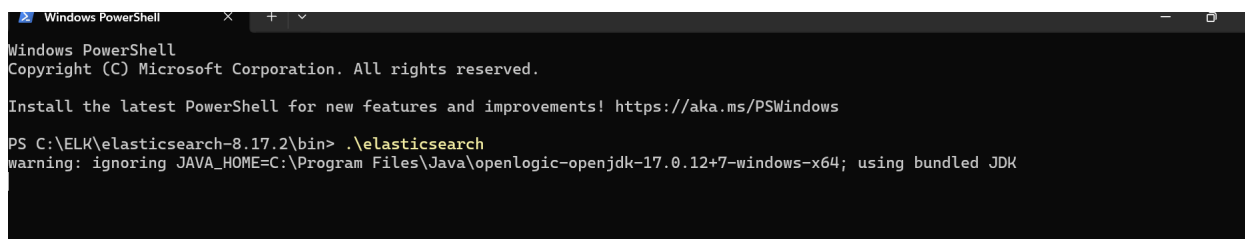
47 # limit.
48 #
49 # Elasticsearch performs poorly when the system is swapping the memory.
50 #
51 # ----- Network -----
52 #
53 # By default Elasticsearch is only accessible on localhost. Set a different
54 # address here to expose this node on the network:
55 #
56 #network.host: 192.168.0.1
57 #
58 # By default Elasticsearch listens for HTTP traffic on the first free port it
59 # finds starting at 9200. Set a specific HTTP port here:
60 #
61 http.port: 9200
62 #
63 # For more information, consult the network module documentation.
64 #
65 # ----- Discovery -----
66 #
67 # Pass an initial list of hosts to perform discovery when this node is started:
68 # The default list of hosts is ["127.0.0.1", "::1"]
69 #
70 #discovery.seed_hosts: ["host1", "host2"]
71 #
72 # Bootstrap the cluster using an initial set of master-eligible nodes:
73 #
74 #cluster.initial_master_nodes: ["node-1", "node-2"]
75 #

Restricted Mode 0 0 0 Ln 62, Col 2 Spaces: 4 UTF-8 CRLF
```

## 2.1.2 Démarrage d'Elasticsearch

Se rendre dans le répertoire d'installation et exécuter :

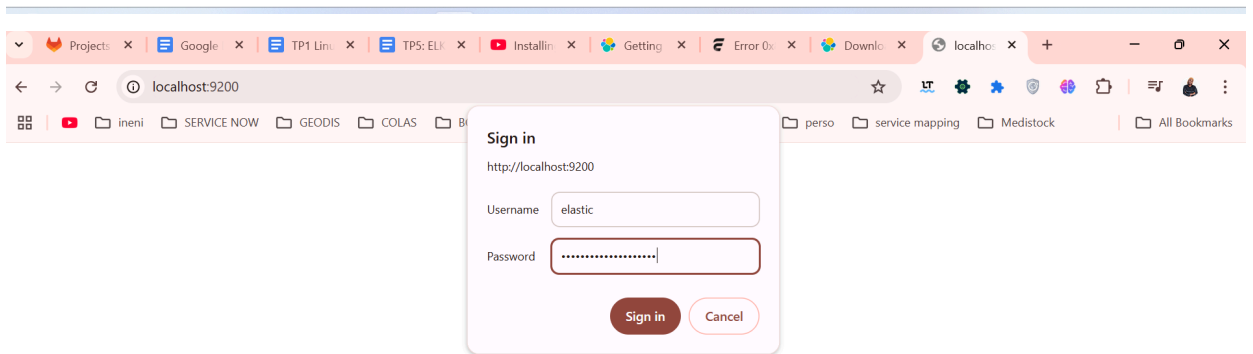
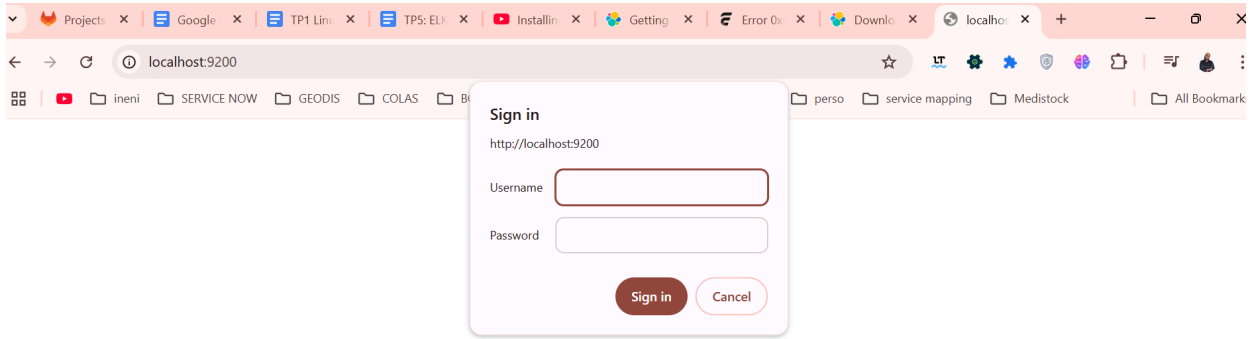
`.\elasticsearch`

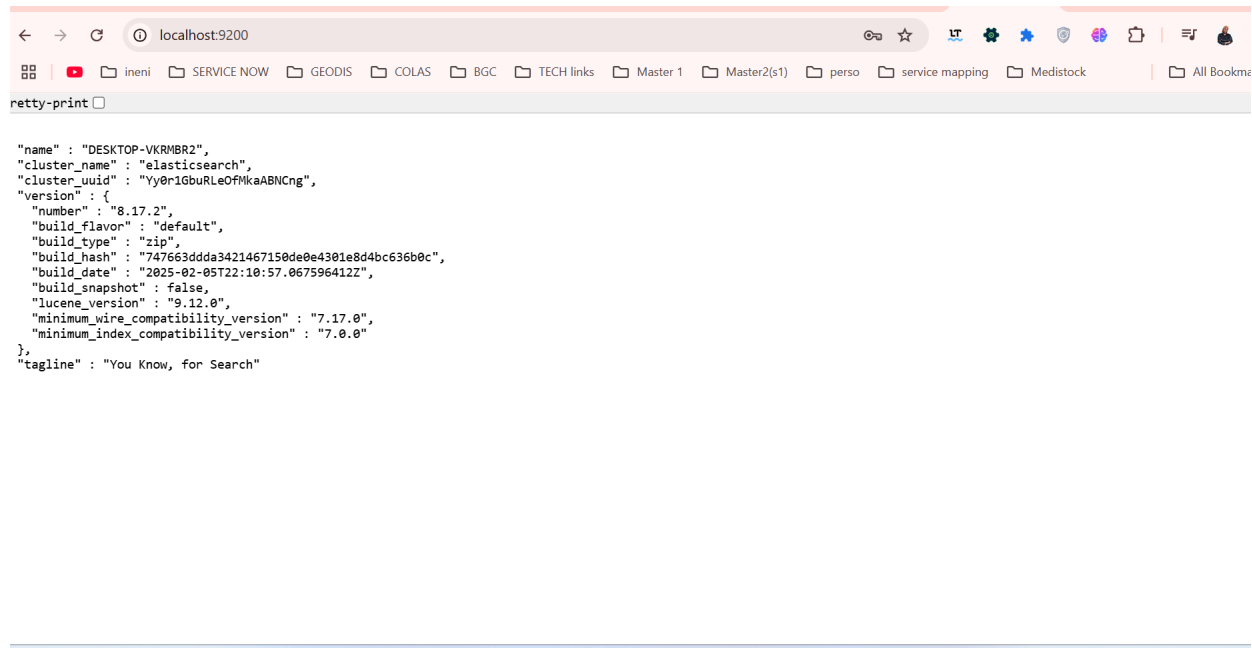


```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\ELK\elasticsearch-8.17.2\bin> .\elasticsearch
warning: ignoring JAVA_HOME=C:\Program Files\Java\openlogic-openjdk-17.0.12+7-windows-x64; using bundled JDK
```





## 2.2 Configuration de Kibana

Kibana est utilisé pour la visualisation des données stockées dans Elasticsearch.

### 2.2.1 Configuration du fichier **kibana.yml**

Modifier le fichier de configuration **config/kibana.yml** :

```
server.port: 5601
server.host: "0.0.0.0"
elasticsearch.hosts: ["http://localhost:9200"]
```

The image shows a Windows environment with two windows. The top window is a code editor displaying the `kibana.yml` configuration file. The file contains settings for the Kibana server, including the server name, SSL configuration, and Elasticsearch connection details. The bottom window is a Windows PowerShell terminal showing the command to start Kibana.

```
! kibana.yml x
C: > ELK > kibana-8.17.2 > config > ! kibana.yml
29 #server.maxPayload: 1048576
30
31 # The Kibana server's name. This is used for display purposes.
32 #server.name: "your-hostname"
33
34 # ===== System: Kibana Server (Optional) =====
35 # Enables SSL and paths to the PEM-format SSL certificate and SSL key files, respectively.
36 # These settings enable SSL for outgoing requests from the Kibana server to the browser.
37 #server.ssl.enabled: false
38 #server.ssl.certificate: /path/to/your/server.crt
39 #server.ssl.key: /path/to/your/server.key
40
41 # ===== System: Elasticsearch =====
42 # The URLs of the Elasticsearch instances to use for all your queries.
43 #elasticsearch.hosts: ["http://localhost:9200"]
44
45 # If your Elasticsearch is protected with basic authentication, these settings provide
46 # the username and password that the Kibana server uses to perform maintenance on the Kibana
47 # index at startup. Your Kibana users still need to authenticate with Elasticsearch, which
48 # is proxied through the Kibana server.
49 #elasticsearch.username: "kibana_system"
50 #elasticsearch.password: "pass"
51
52 # Kibana can also authenticate to Elasticsearch via "service account tokens".
53 # Service account tokens are Bearer style tokens that replace the traditional username/password based configuration.
54 # Use this token instead of a username/password.
55 #elasticsearch.serviceAccountToken: "my_token"
56

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\ELK\kibana-8.17.2\bin> .\kibana.bat
```

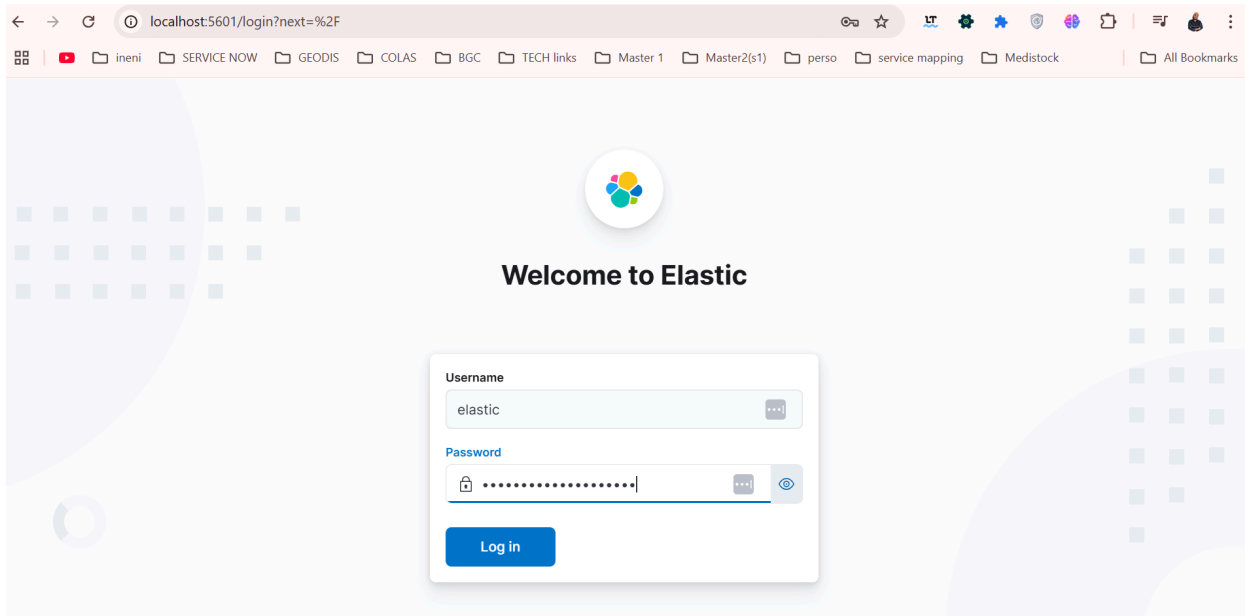
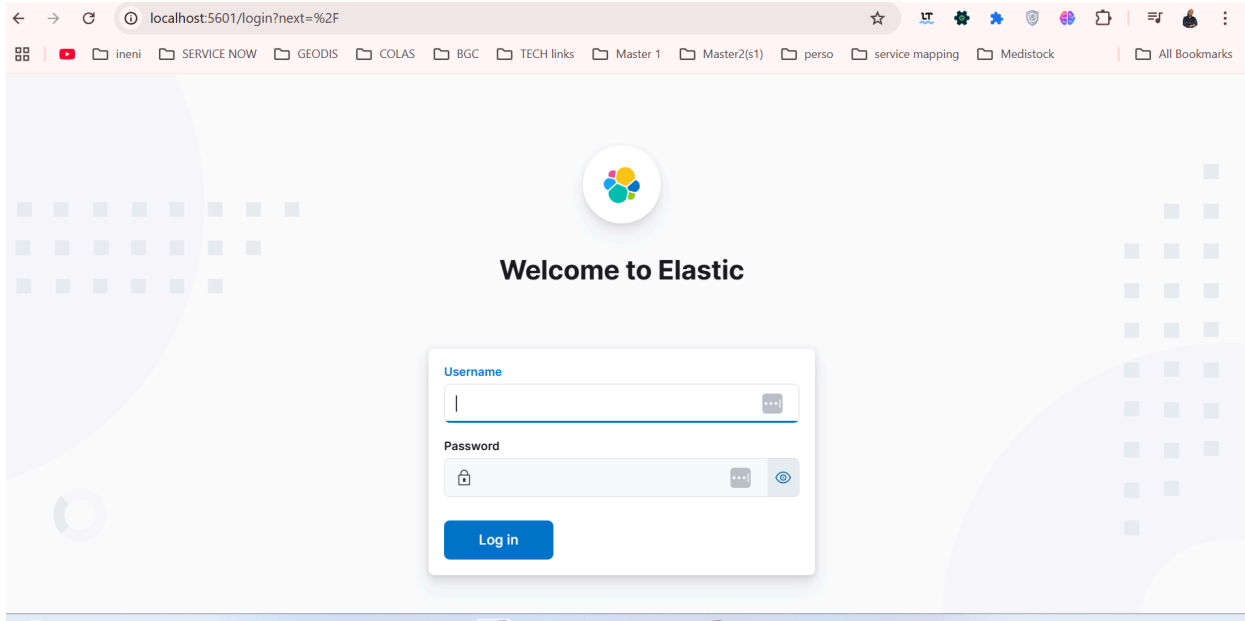
## 2.2.2 Démarrage de Kibana

Depuis le répertoire d'installation, exécuter :

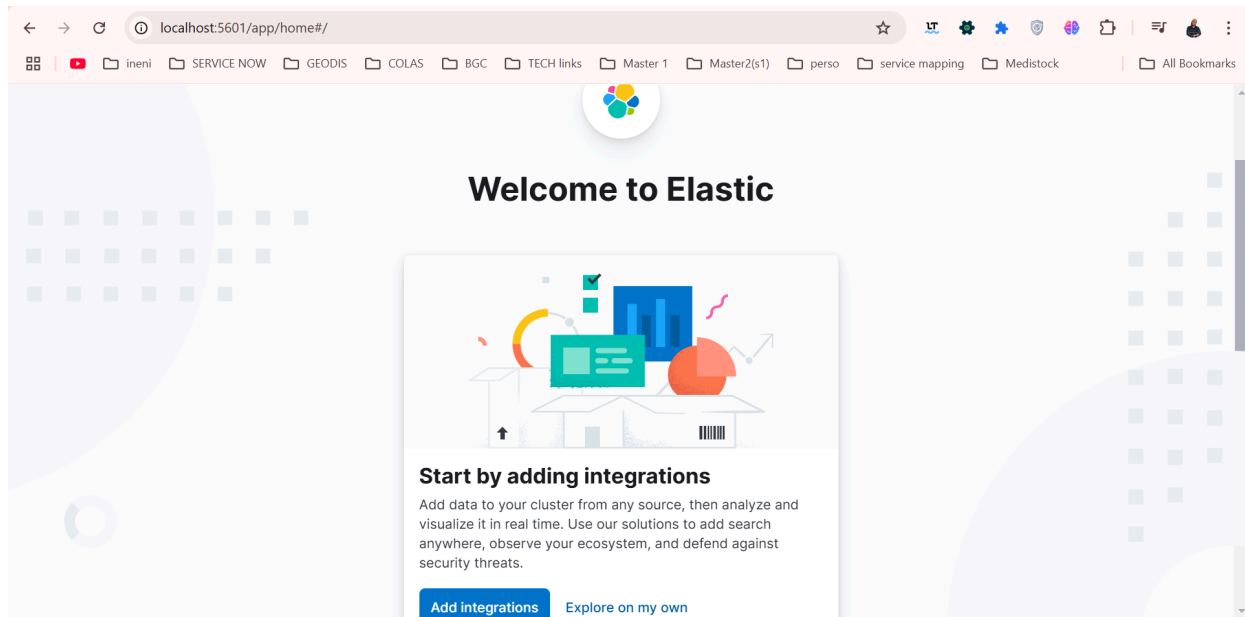
`.\kibana.bat`

Accéder à l'interface web via :

<http://localhost:5601>







## 2.3 Configuration de Logstash

Logstash est le composant qui collecte et envoie les logs vers Elasticsearch.

### 2.3.1 modification du fichier **logstash.conf**

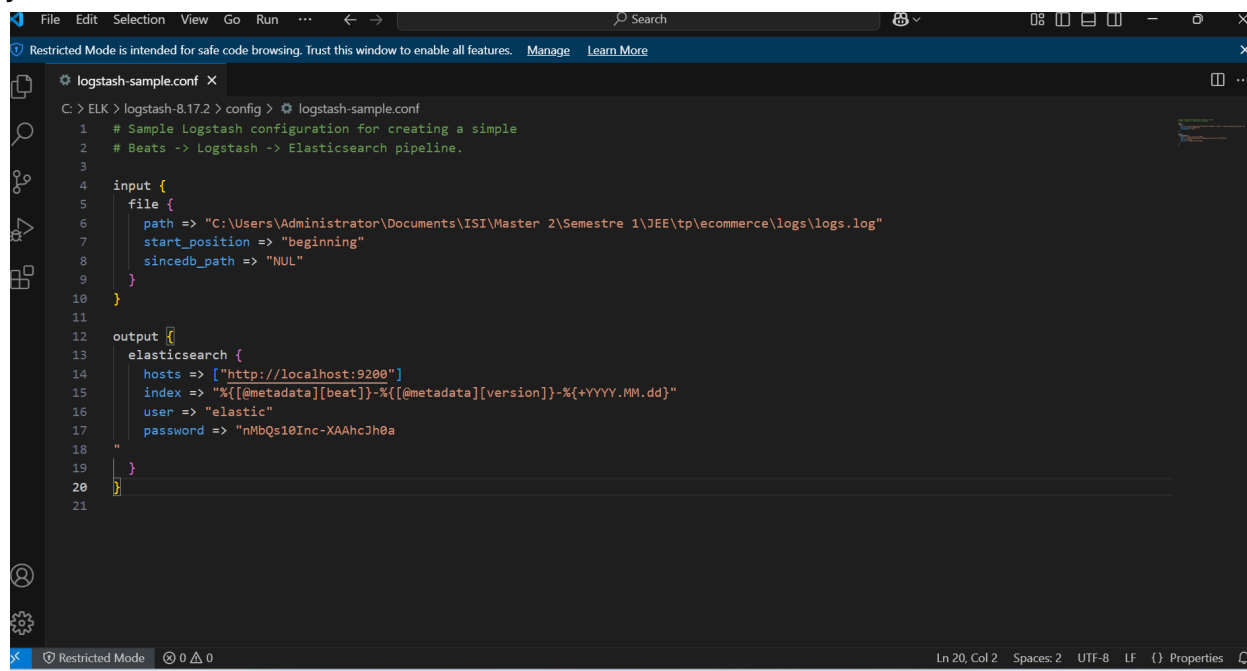
modifier le fichier de configuration **logstash.conf** :

```
input {  
  file {  
    path => "/chemin/vers/logs.log"  
    start_position => "beginning"  
    sincedb_path => "NUL"  
    ignore_older => 0  
  }  
}  
  
output {  
  elasticsearch {  
    hosts => ["http://localhost:9200"]  
  }  
}
```

```

index => "logstash-logs-%{+YYYY.MM.dd}"
user => "elastic"
password => "changeme"
}
stdout { codec => rubydebug }
}

```



## 2.3.2 Démarrage de Logstash

Exécuter Logstash avec la configuration créée :

`.\logstash -f logstash.conf`



Vérifier que Logstash envoie bien les logs à Elasticsearch.

localhost:5601/app/management/data/index\_management/indices

elastic

Find apps, content, and more.

Stack Management Index Management Indices

## Index Management

[Index Management docs](#)

[Indices](#) Data Streams Index Templates Component Templates Enrich Policies

Update your Elasticsearch indices individually or in bulk. [Learn more.](#) ☐ Include hidden indices ☐ Include rollout indices

Search Lifecycle status Lifecycle phase Reload indices Create index

| Name                          | Health | Status | Primaries | Replicas | Docume... | Storage ... | Data stre... |
|-------------------------------|--------|--------|-----------|----------|-----------|-------------|--------------|
| logstash-ecommerce-2025.02.22 | yellow | open   | 1         | 1        | 827       | 567.74kb    |              |

Rows per page: 10

Console Notebooks

localhost:5601/app/management/data/index\_management/indices/index\_details?indexName=logstash-ecommerce...

elastic

Find apps, content, and more.

Stack Management Index Management Indices Index details Overview

## logstash-ecommerce-2025.02.22

[Back to indices](#) [Manage index](#) [Discover index](#)

[Overview](#) Mappings Settings Statistics

**Storage**

567.74kb Primary  
567.74kb Total

Shards 1 Primary / 1 Replica

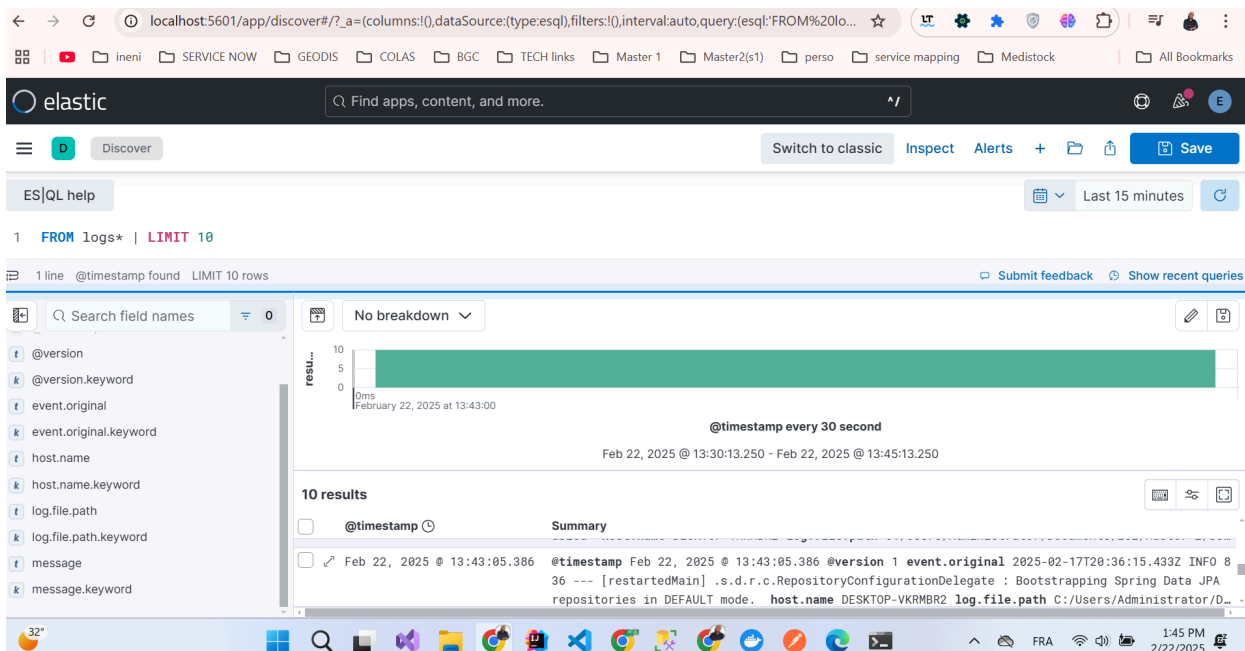
**Status**

Open Warning

827 Documents / 0 Deleted

**Add data to this index**

Console Notebooks



et voir aussi les logs sur le terminal

```

Windows PowerShell
}
},
  "host" => {
    "name" => "DESKTOP-VKRM2"
  },
  "@timestamp" => 2025-02-22T13:44:07.546301500Z,
  "event" => {
    "original" => "[http-nio-8080-exec-6] DEBUG | org.springframework.web.servlet.mvc.method.annotation.HttpEntityMethodProcessor
| Using 'application/json', given [*/] and supported [application/json, application/*+json] (AbstractMessageConverterMethodProcess
r.java:310)\r"
  }
}
{
  "@version" => "1",
  "message" => "[http-nio-8080-exec-6] ERROR | group.isi.ecommerce.config.LoggingAspect | Exception in group
.isi.ecommerce.products.controllers.ProductController.getProduct() with message = Requested Roles with ref = ref6r does not exist (Lo
ggingAspect.java:34)\r",
  "log" => {
    "file" => {
      "path" => "C:/Users/Administrator/Documents/ISI/Master 2/Semestre 1/JEE/tp/ecommerce/logs/logs.log"
    }
  },
  "host" => {
    "name" => "DESKTOP-VKRM2"
  },
  "@timestamp" => 2025-02-22T13:44:07.545264800Z,
  "event" => {
    "original" => "[http-nio-8080-exec-6] ERROR | group.isi.ecommerce.config.LoggingAspect | Exception in gro
up.isi.ecommerce.products.controllers.ProductController.getProduct() with message = Requested Roles with ref = ref6r does not exist (
LoggingAspect.java:34)\r"
  }
}
}

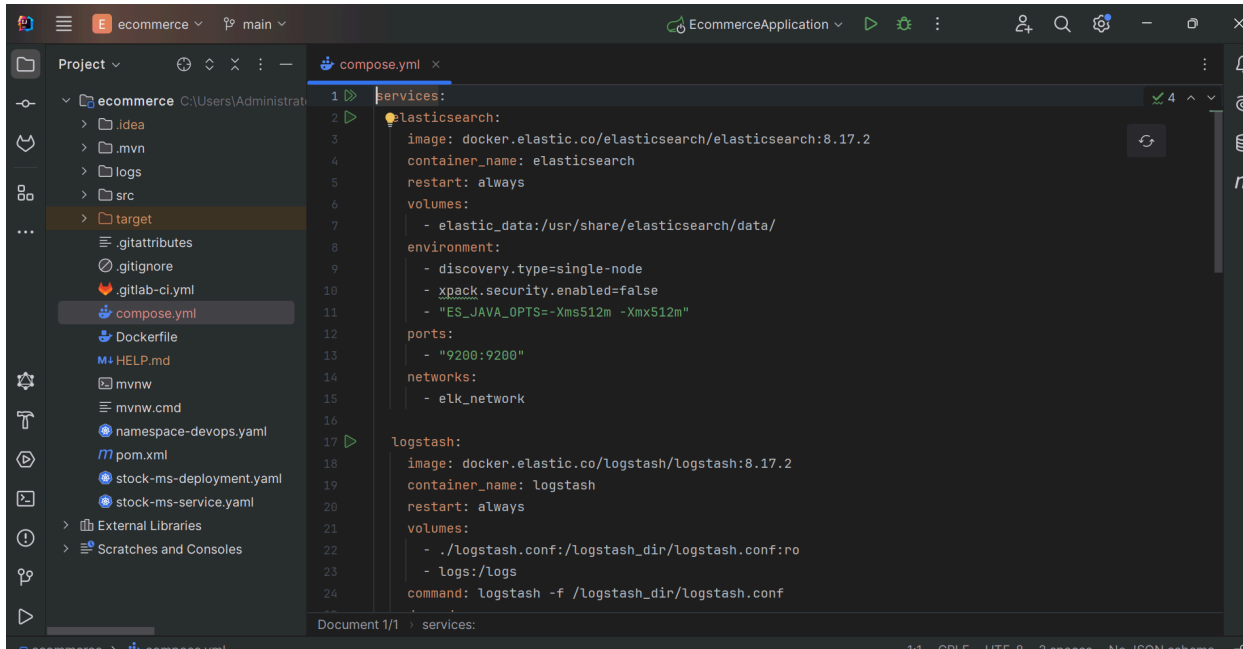
```

### 3. Mise en place de la stack ELK avec Docker Compose

L'utilisation de **Docker Compose** permet d'automatiser le déploiement et de faciliter la gestion des services.

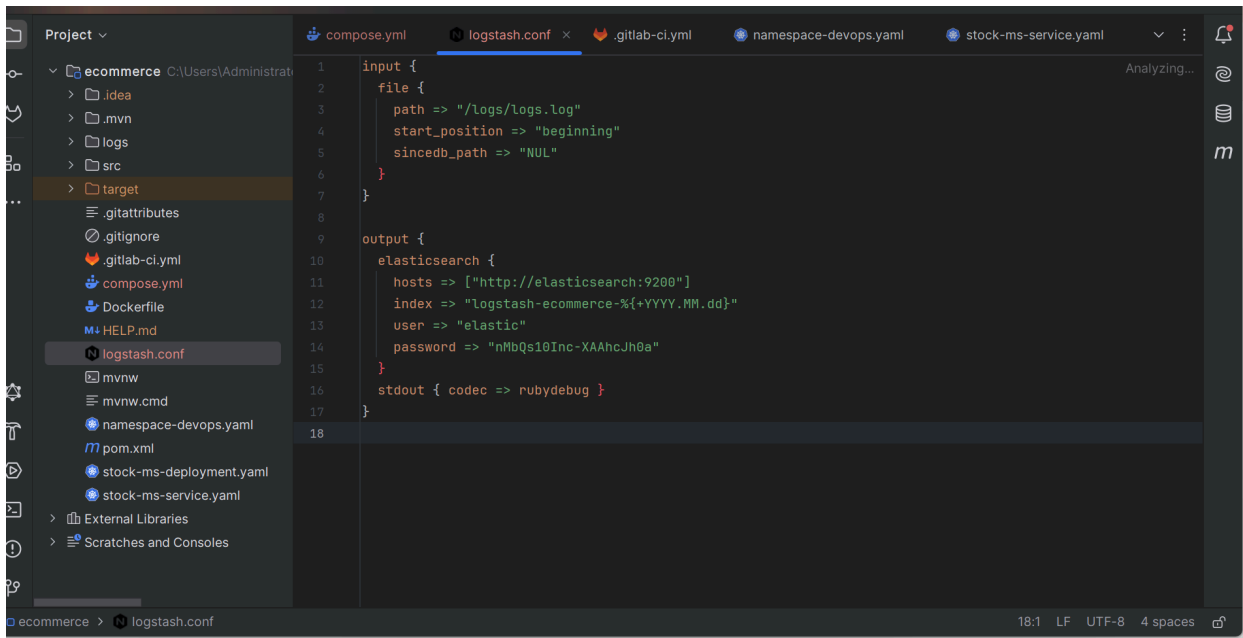
### 3.1 Création des fichiers `docker-compose.yml` et `logstash.conf`

Ces fichiers définissent les services Elasticsearch, Logstash, Kibana et l'application Stock-MS.



```
services:
  elasticsearch:
    image: docker.elastic.co/elasticsearch/elasticsearch:8.17.2
    container_name: elasticsearch
    restart: always
    volumes:
      - elastic_data:/usr/share/elasticsearch/data/
    environment:
      - discovery.type=single-node
      - xpack.security.enabled=false
      - "ES_JAVA_OPTS=-Xms512m -Xmx512m"
    ports:
      - "9200:9200"
    networks:
      - elk_network

  logstash:
    image: docker.elastic.co/logstash/logstash:8.17.2
    container_name: logstash
    restart: always
    volumes:
      - ./logstash.conf:/logstash_dir/logstash.conf:ro
      - logs:/logs
    command: logstash -f /logstash_dir/logstash.conf
```



```
input {
  file {
    path => "/logs/logs.log"
    start_position => "beginning"
    sincedb_path => "NUL"
  }
}

output {
  elasticsearch {
    hosts => ["http://elasticsearch:9200"]
    index => "logstash-ecommerce-%{+YYYY.MM.dd}"
    user => "elastic"
    password => "nMbQs10Inc-XAAhcJh0a"
  }
  stdout { codec => rubydebug }
}
```

## 4. Intégration de Stock-MS avec ELK

L'application **Stock-MS** a été configurée pour envoyer ses logs à **Logstash**, qui les transfère ensuite vers Elasticsearch.

## 4.1 Modification de `application.properties` dans Stock-MS

Ajout de la configuration pour les logs :

`logging.file.name=/logs/logs.log`

Cela permet d'écrire les logs dans un fichier partagé avec Logstash.

## 5. Démarrage et tests de la stack ELK

Démarrer la stack :

`docker-compose up -d`

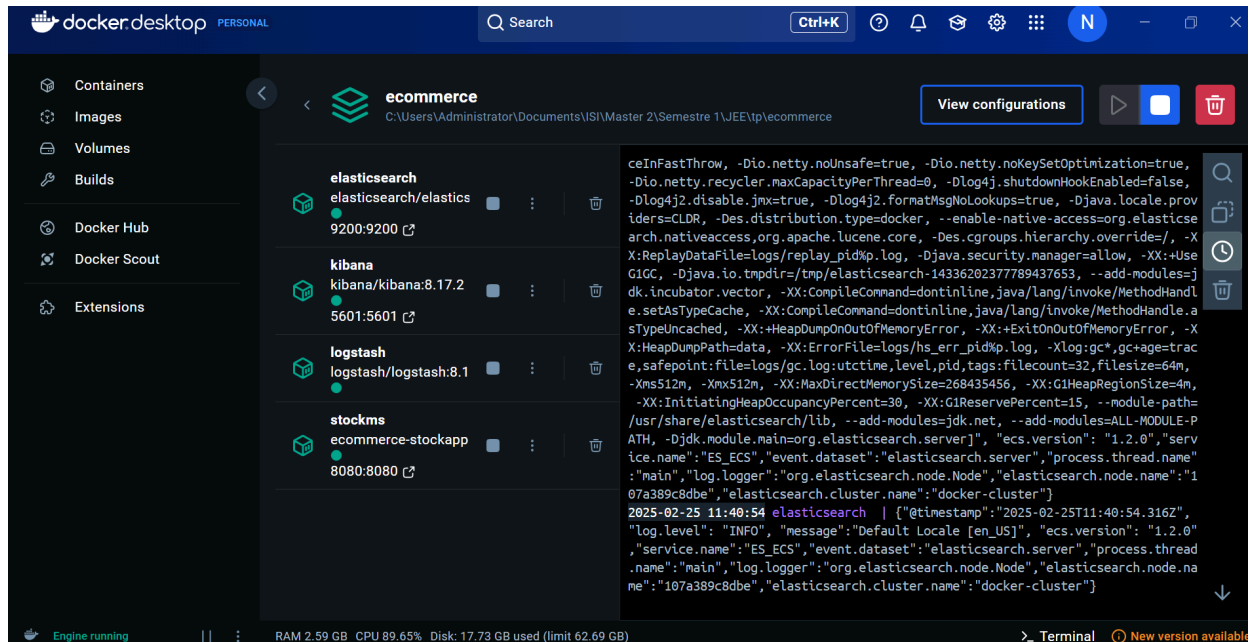
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

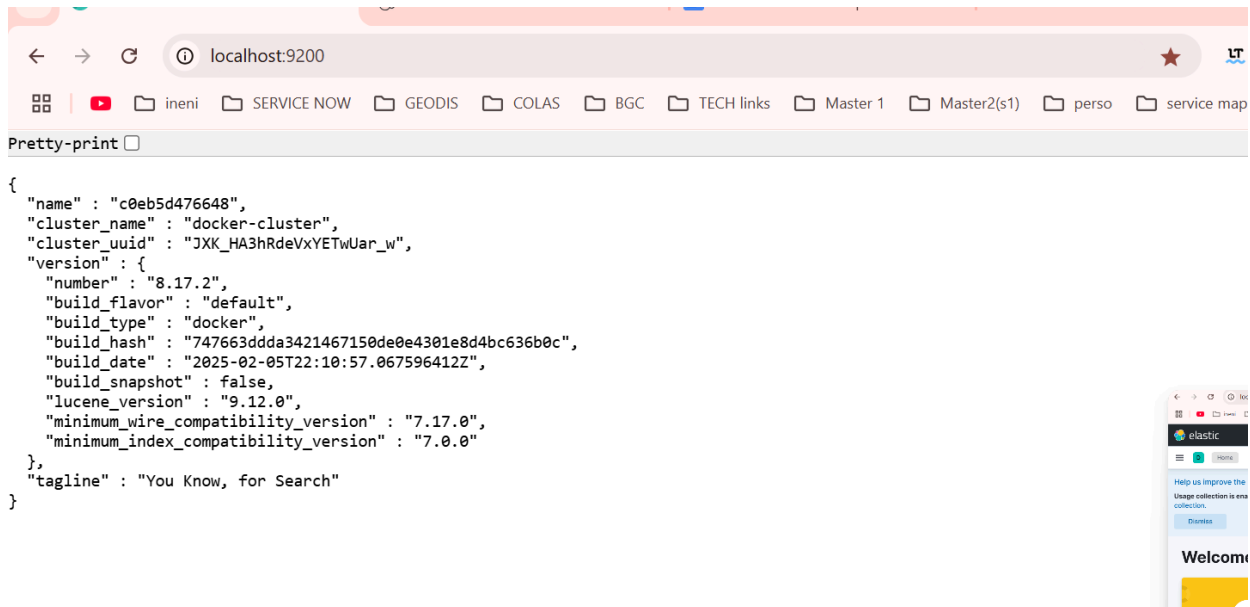
PS C:\Users\Administrator\Documents\ISI\Master 2\Semestre 1\JEE\tp\e-commerce> docker compose up -d
[+] Running 0/3
 - elasticsearch Pulling      6.7s
 - kibana Pulling            6.7s
 - logstash Pulling          6.7s
```

```
+ Building 0.0s (0/0) docker:desktop-linux
+ Running 0/1kapp Building
+ Building 20.3s (9/9) FINISHED
=> [stockapp internal] load build definition from Dockerfile
=> => transferring dockerfile: 227B
=> [stockapp internal] load metadata for docker.io/library/openjdk:17-jdk-slim
=> [stockapp auth] library/openjdk:pull token for registry-1.docker.io
=> [stockapp internal] load .dockerignore
=> => transferring context: 2B
=> [stockapp internal] load build context
=> => transferring context: 59.46MB
=> CACHED [stockapp 1/2] FROM docker.io/library/openjdk:17-jdk-slim@sha256:aaa3b3cb27e3e520b8f116863d0580c438ed55ecfa0bc126b4
=> => resolve docker.io/library/openjdk:17-jdk-slim@sha256:aaa3b3cb27e3e520b8f116863d0580c438ed55ecfa0bc126b41f68c3f62f9774
=> [stockapp 2/2] COPY target/stock-ms.jar stock-ms.jar
=> [stockapp] exporting to image
=> => exporting layers
=> => exporting manifest sha256:68041fddbc52779515386fad6829d5e87b951d7a87953a0effald3413f10d762
=> => exporting config sha256:abcd8f15d1f71f7b1034b4a6779f9810720cdb46257ea21e1ef8bd13cee025e0
=> => exporting attestation manifest sha256:324d1ed76ad7f5798e6dbb7761e83dc88228fa7183cbcd95d2caaf68e4768e9a
=> => exporting manifest list sha256:7404d65ae5ecd3fe30c64dc2f7c48a00a2893fb2252f59f1ce325b4430d1476b
=> => naming to docker.io/library/e-commerce-stockapp:latest
+ Running 6/7g to docker.io/library/e-commerce-stockapp:latest
✔ Service stockapp Built 20.8s
✔ Network ecommerce_elk_network Created 0.1s
✔ Volume "ecommerce_elastic_data" Created 0.0s
✔ Volume "ecommerce_logs" Created 0.0s
✔ Container elasticsearch Created 0.3s
- Container logstash Creating 3.1s
✔ Container kibana Created 0.2s
error during connect: Post "http://%2F%2F.%2Fpipe%2FdockerDesktopLinuxEngine/v1.47/containers/create?name=logstash": EOF
S C:\Users\Administrator\Documents\ISI\Master 2\Semestre 1\JEE\tp\e-commerce> |
```

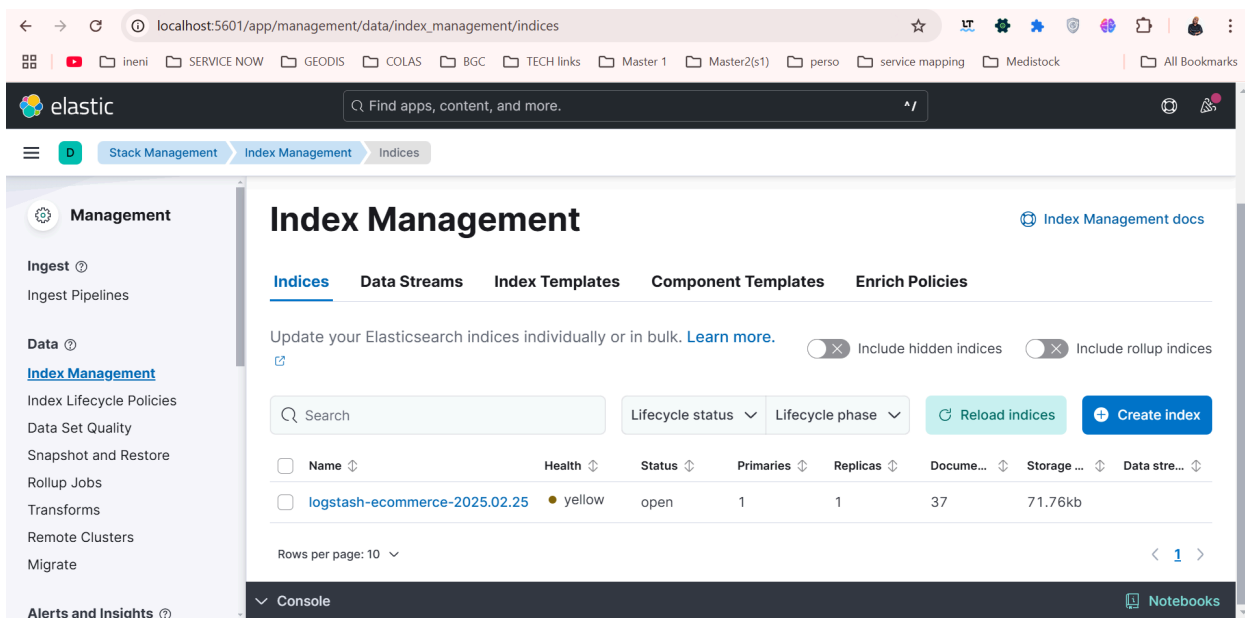
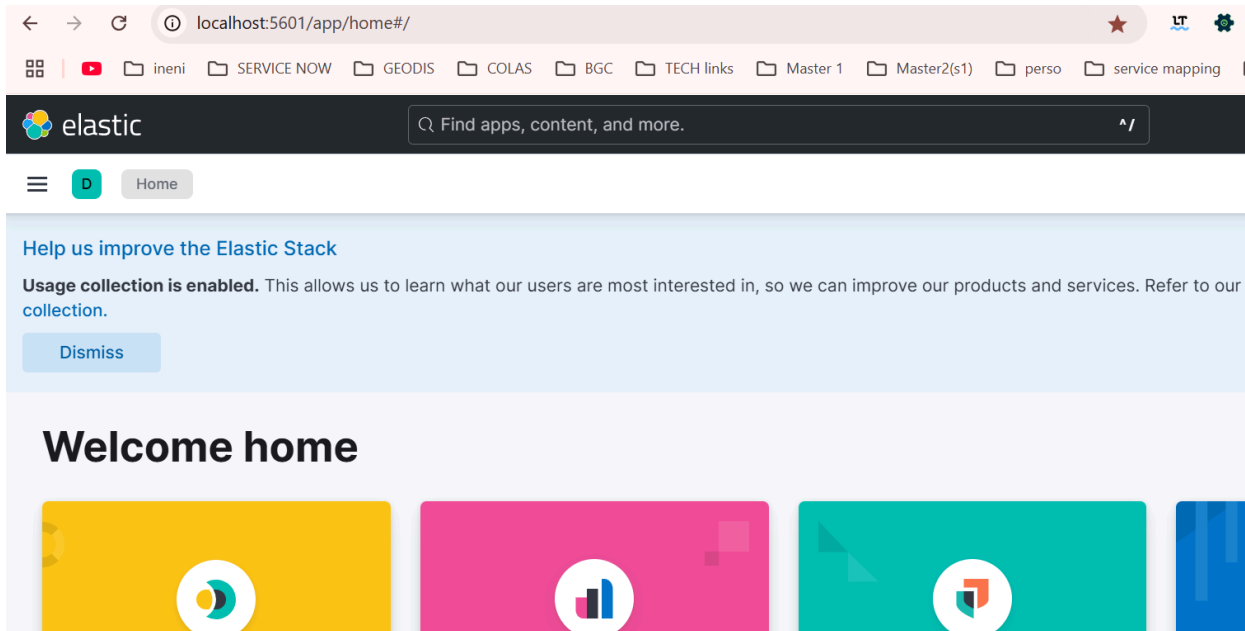
## 1. Vérifier que tous les conteneurs fonctionnent :



## 2. Tester Elasticsearch :



## 3. Tester Kibana en accédant à <http://localhost:5601>



1

## Conclusion

---

<sup>1</sup> Seynabou SOUMARE M2 GL



La mise en place de la stack ELK permet d'assurer une collecte centralisée des logs et leur visualisation via Kibana. L'intégration avec Docker Compose facilite l'automatisation et la gestion des services, permettant ainsi un déploiement rapide et efficace.