

# DevOps Project

ICCN2

*Realized by:*

MIFOUAJ Mounssif

LAOUIJ Hamza

LAAFOU Anasse

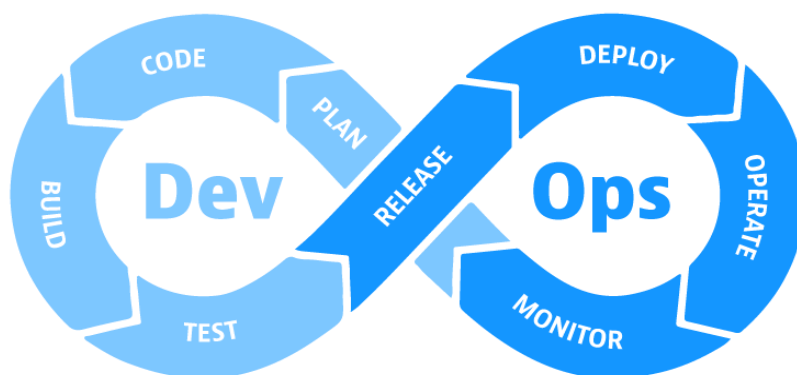
BENTAJA Othmane

ZRAIDI Meriem

AIT-IZANA Ismail

*Supervised by:*

Mr ALLAKI Driss



2021-2022

## Team Topology

KATA is an organization with a single main web-based product, so we chose “Fully Shared Ops Responsibilities” topology where operations people have been integrated in product development teams.

There is so little separation between Dev and Ops that all people are highly focused on a shared purpose.



Scrum is an Agile software development framework that enables a team to communicate and self-organize.

The SCRUM team is composed of:

**Product owner:** responsible for the project's outcome and seeks to maximize a product's value by managing and optimizing the product backlog

**The scrum master:** helps the team enhance and streamline the processes by which they achieve their goals.

**Team members:** carry out all work required to build increments of value every sprint, and can include researchers, architects, designers ...

However, the SCRUM workflow is composed of:

**Sprint:** is the basic unit of development in scrum, its length is agreed and fixed in advance for each sprint and is normally between one week and one month, with two weeks being the most common.

**Sprint planning:** The purpose of sprint planning is to define what can be delivered in the sprint and how that work will be achieved.

**Sprint Review:** one of the most important ceremonies in Scrum where the team gathers to review completed work and determine whether additional changes are needed.

**Sprint Retrospective:** a recurring meeting held at the end of a sprint used to discuss what went well during the previous sprint cycle and what can be improved for the next sprint.

To manage work in SCRUM, we use the following documents:

**Product Backlog:** a prioritized list of work for the development team that is derived from the roadmap and its requirements. The most important items are shown at the top of the product backlog so the team knows what to deliver first.

**Sprint Backlog:** a list of tasks identified by the Scrum team to be completed during the Scrum sprint. During the sprint planning meeting, the team selects some number of product backlog items, usually in the form of user stories, and identifies the tasks necessary to complete each user story.

**User Story:** an informal, general explanation of a software feature written from the perspective of the end user or customer. They don't go into detail. Requirements are added later, once agreed upon by the team.

**Story Points:** units of measure for expressing an estimate of the overall effort required to fully implement a product backlog item.

**Definition of Done:** represents the organization's formal definition of quality for all Product Backlog Items.

**The Product Increment:** the sum of all the Product Backlog items completed during a Sprint and the value of the increments of all previous Sprints.

To manage our agile project, we chose to use JIRA as our agile project management tool:

We divided our roadmap to 5 epics:

1- Web Development

2- Clientui Pipeline

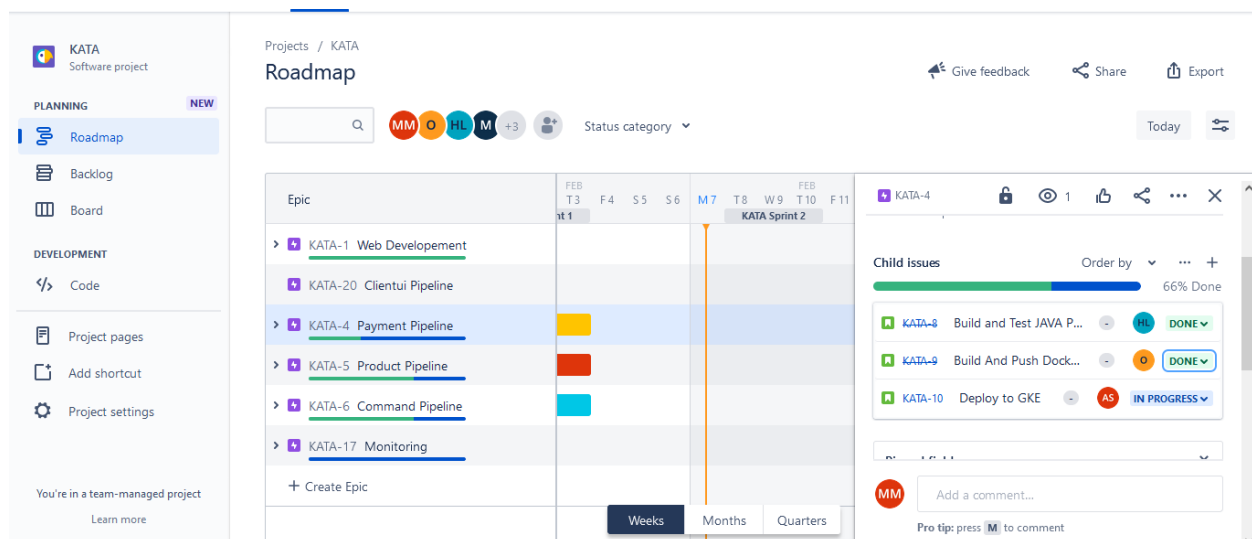
3- Command Pipeline

4- Product Pipeline

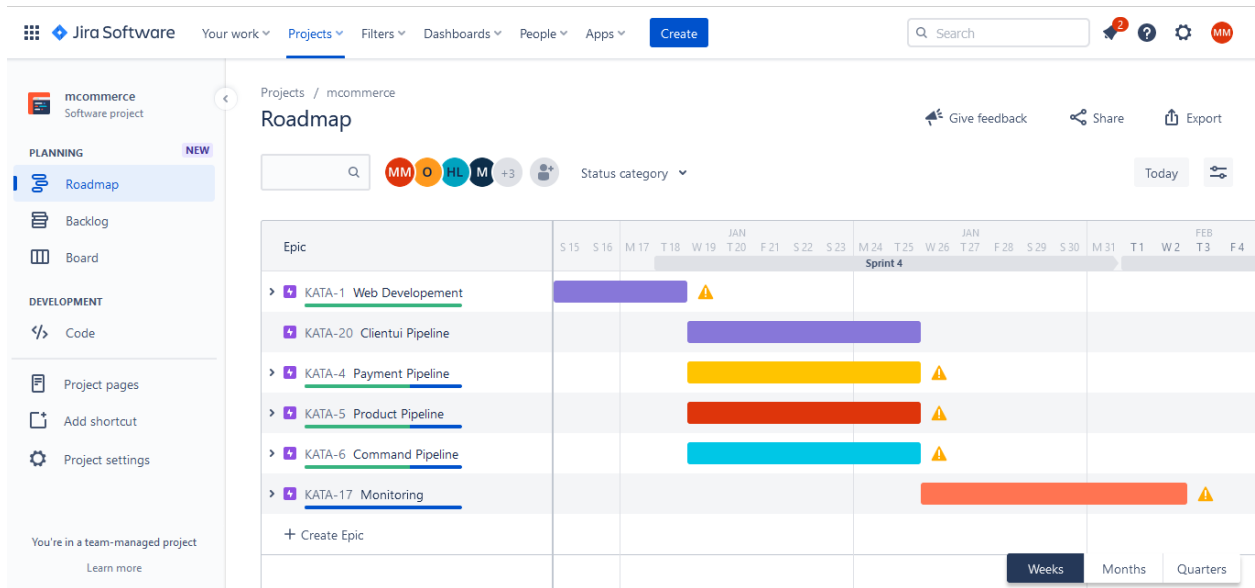
5- Payment Pipeline

6- Monitoring

In the image below, we created the jobs and the tasks in each jobs:

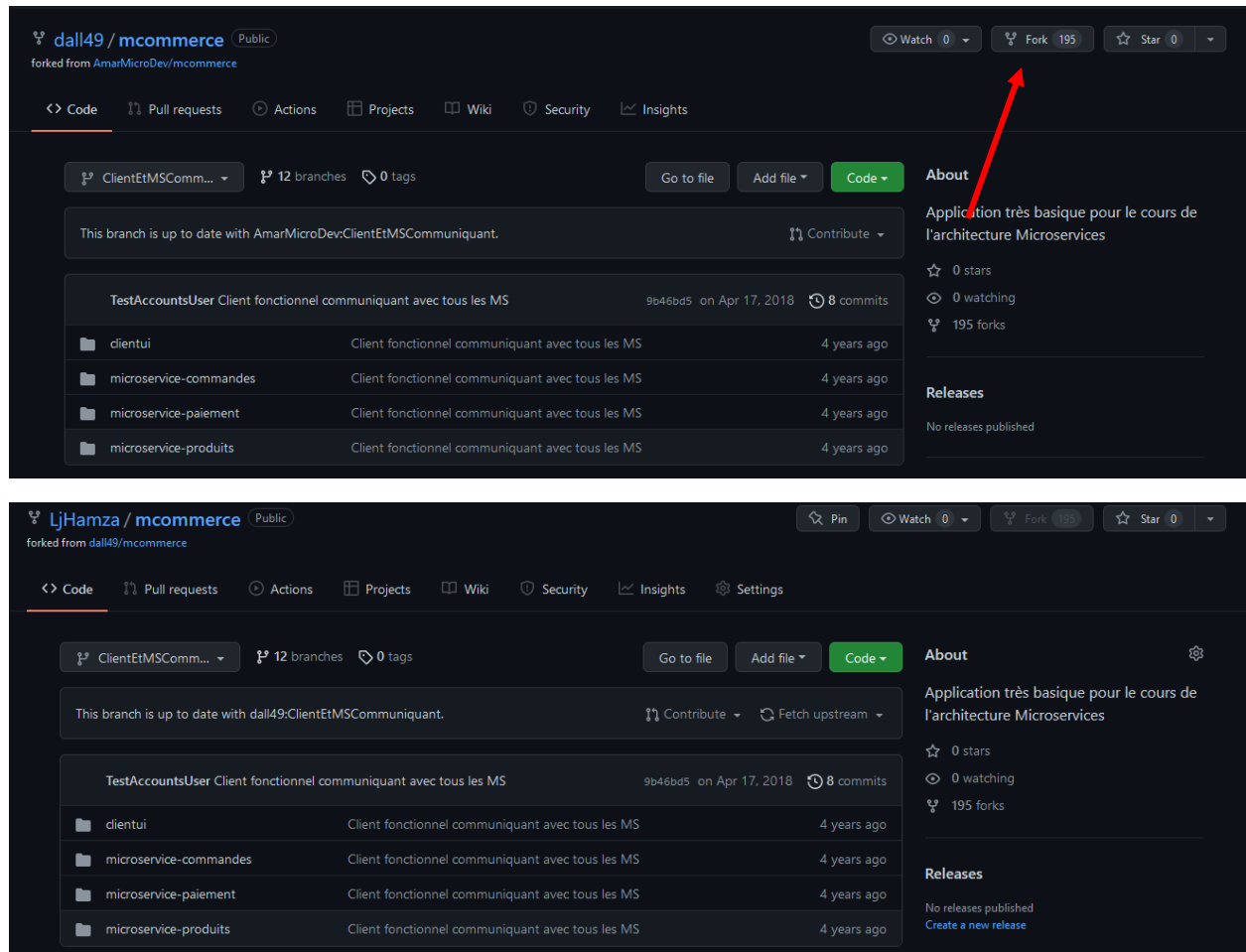


Finally, the image below shows an overview of the project management:

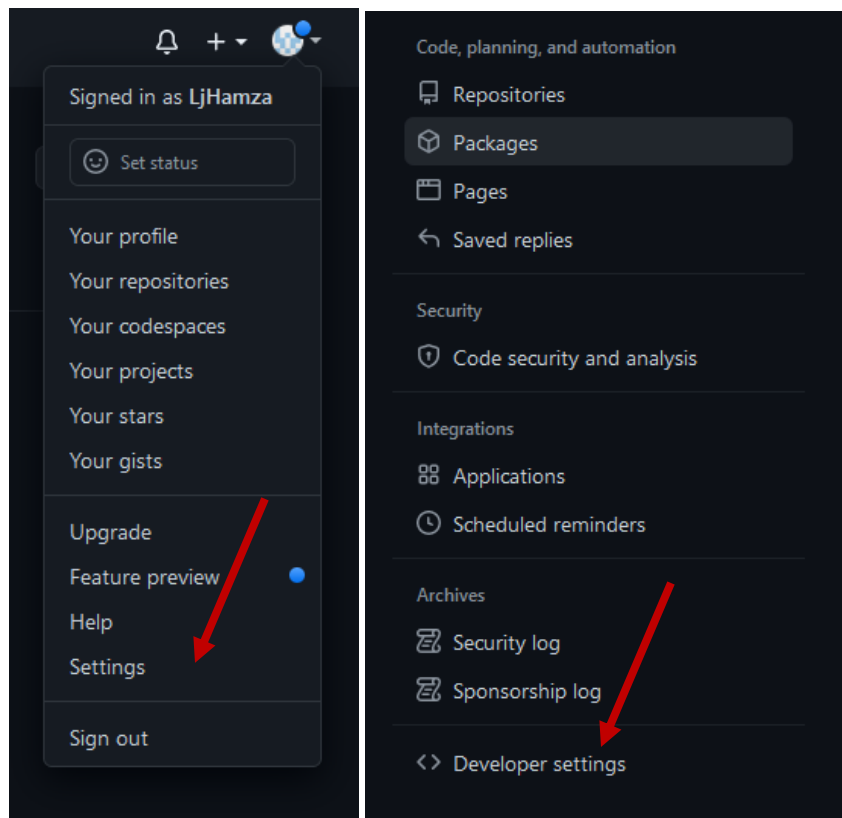


# Fork Project from GitHub

First, we will fork the project:



Then, we will create token to connect GitHub with Gitlab.



GitHub Apps

OAuth Apps

Personal access tokens

## New personal access token

Personal access tokens function like ordinary OAuth access tokens. They can be used instead of a password for Git over HTTPS, or can be used to authenticate to the API over Basic Authentication.

**Note**

GitLab Token

What's this token for?

**Expiration \***

30 days

The token will expire on Thu, Feb 24 2022

**Select scopes**

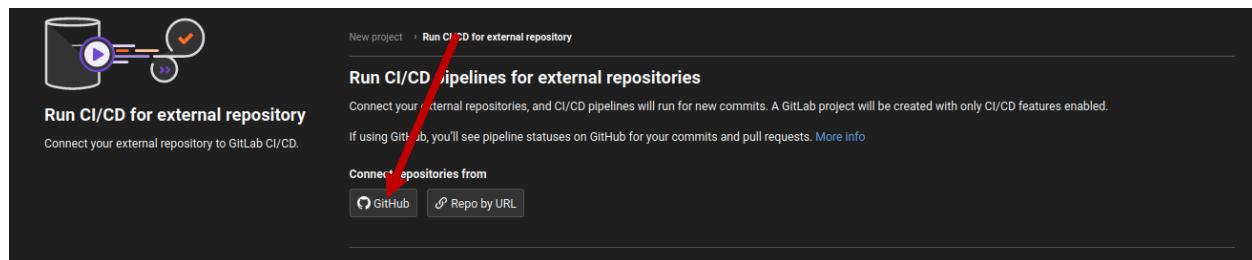
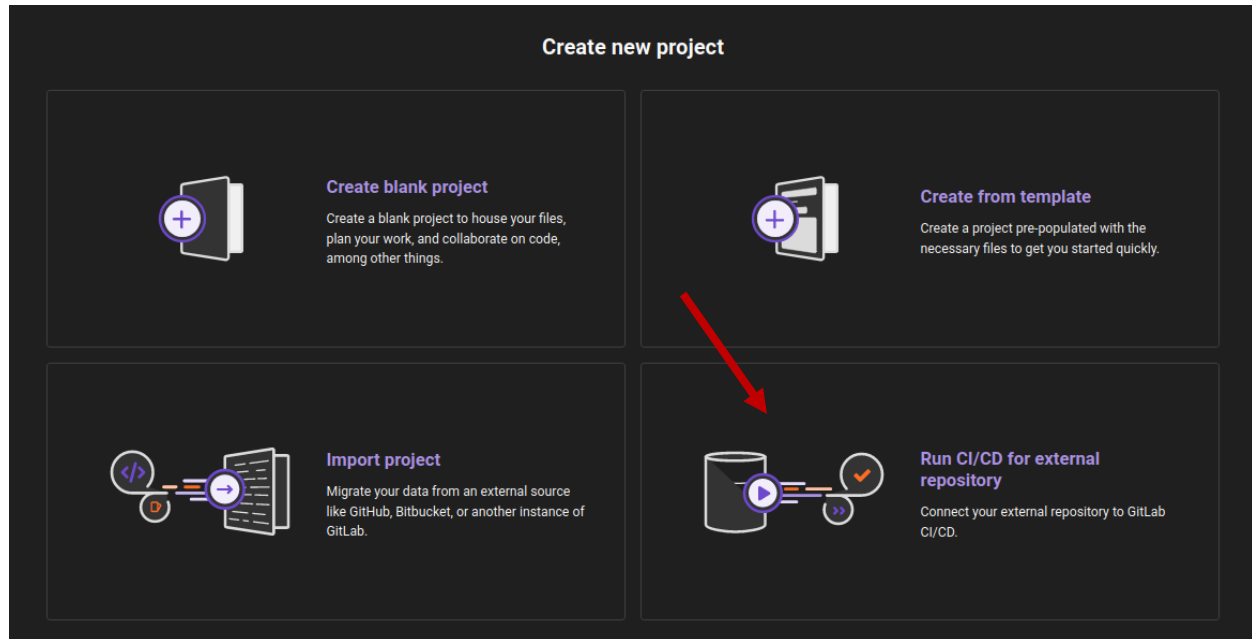
Scopes define the access for personal tokens. [Read more about OAuth scopes.](#)

<input checked="" type="checkbox"/> repo	Full control of private repositories
<input checked="" type="checkbox"/> repo:status	Access commit status
<input checked="" type="checkbox"/> repo_deployment	Access deployment status
<input checked="" type="checkbox"/> public_repo	Access public repositories
<input checked="" type="checkbox"/> repo:invite	Access repository invitations
<input checked="" type="checkbox"/> security_events	Read and write security events

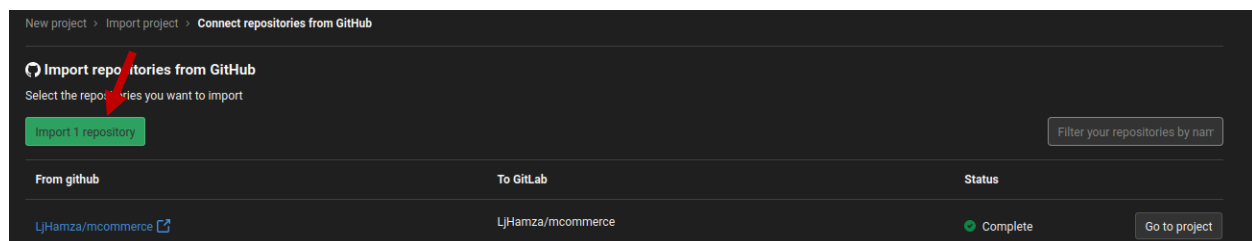
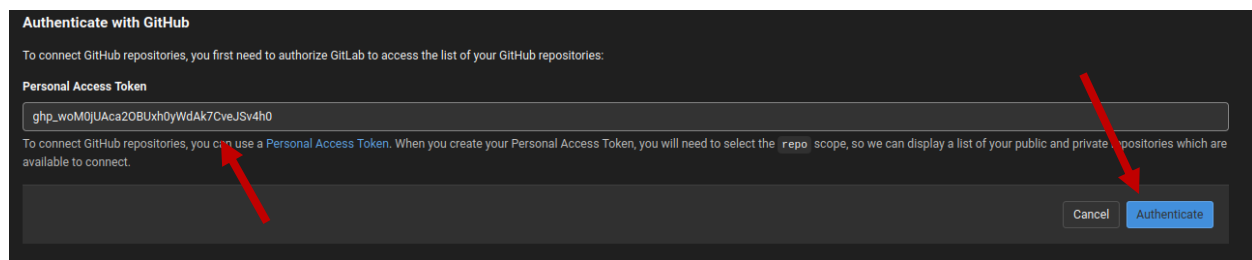
<input checked="" type="checkbox"/>	<b>admin:repo_hook</b>	Full control of repository hooks
<input checked="" type="checkbox"/>	write:repo_hook	Write repository hooks
<input checked="" type="checkbox"/>	read:repo_hook	Read repository hooks
<input type="checkbox"/>	<b>admin:org_hook</b>	Full control of organization hooks
<input type="checkbox"/>	<b>gist</b>	Create gists



# Create Gitlab Project

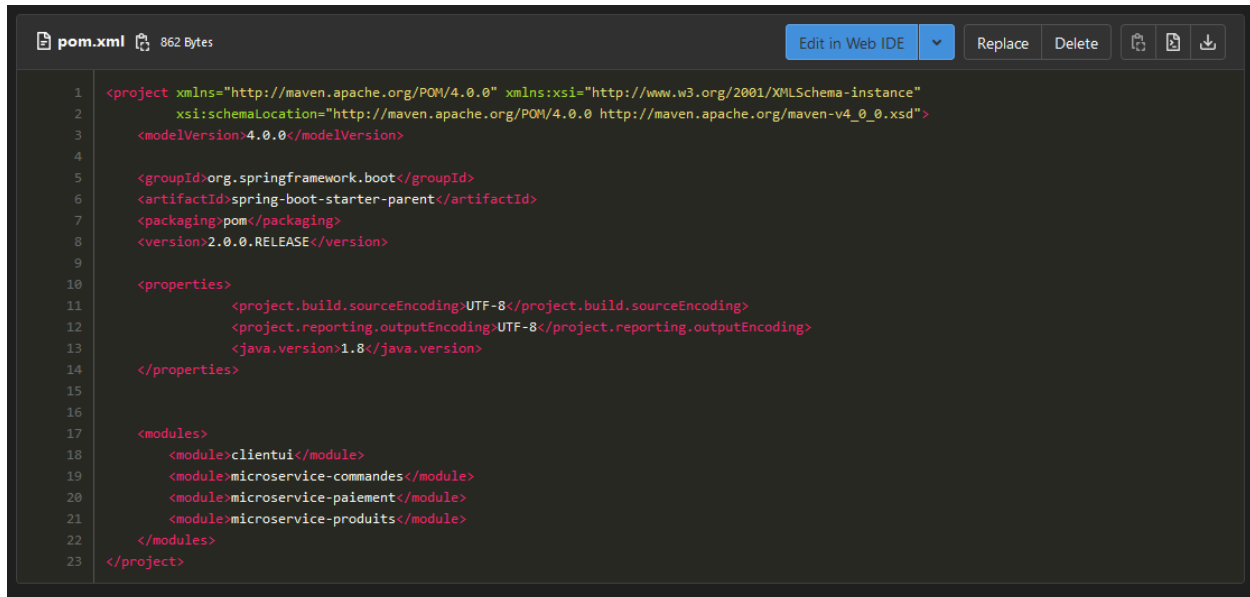


We paste the Github token:



# Create Pipeline

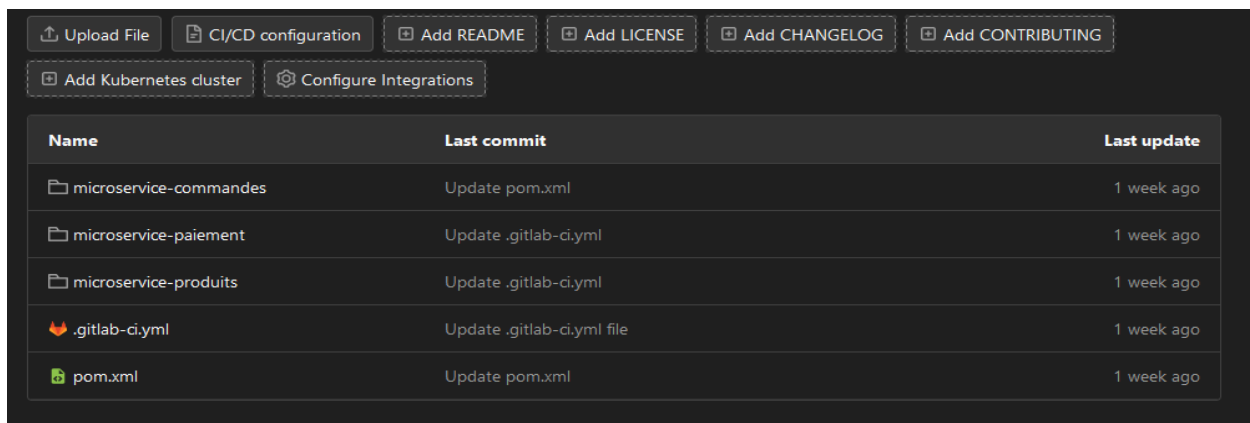
Create pom.xml file in root directory



The screenshot shows a web IDE interface with a file named `pom.xml` (862 Bytes). The code is as follows:

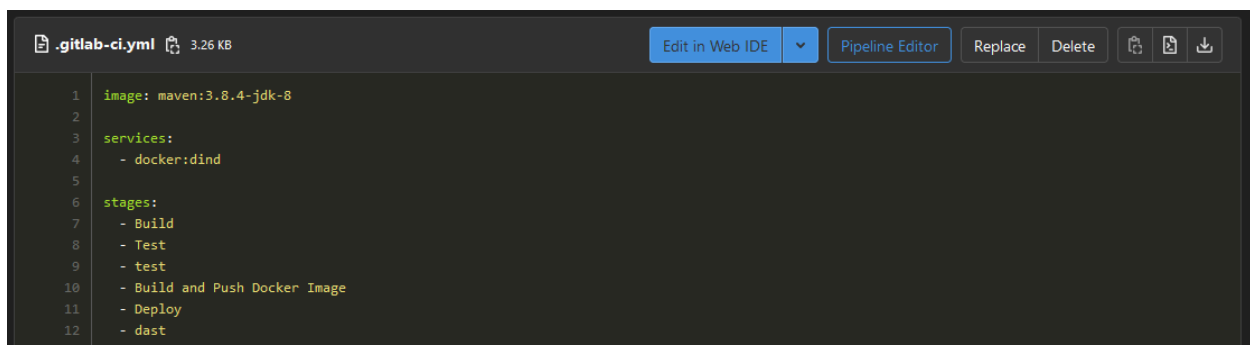
```
1 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
3   <modelVersion>4.0.0</modelVersion>
4
5   <groupId>org.springframework.boot</groupId>
6   <artifactId>spring-boot-starter-parent</artifactId>
7   <packaging>pom</packaging>
8   <version>2.0.0.RELEASE</version>
9
10  <properties>
11    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
12    <project.reporting.outputEncoding>UTF-8</project.reporting.outputEncoding>
13    <java.version>1.8</java.version>
14  </properties>
15
16  <modules>
17    <module>clientui</module>
18    <module>microservice-commandes</module>
19    <module>microservice-paiement</module>
20    <module>microservice-produits</module>
21  </modules>
22 </project>
```

Create .gitlab-ci.yml file for the pipeline configuration in root directory



The screenshot shows the GitLab repository interface with a table of file history. The table has three columns: Name, Last commit, and Last update.

Name	Last commit	Last update
microservice-commandes	Update pom.xml	1 week ago
microservice-paiement	Update .gitlab-ci.yml	1 week ago
microservice-produits	Update .gitlab-ci.yml	1 week ago
.gitlab-ci.yml	Update .gitlab-ci.yml file	1 week ago
pom.xml	Update pom.xml	1 week ago



The screenshot shows a web IDE interface with a file named `.gitlab-ci.yml` (3.26 KB). The code is as follows:

```
1 image: maven:3.8.4-jdk-8
2
3 services:
4   - docker:dind
5
6 stages:
7   - Build
8   - Test
9   - test
10  - Build and Push Docker Image
11  - Deploy
12  - dast
```

## Build Stage

To keep our code clean we created build template and a variable named MODULE which we will pass to it the name of the four services.

```
variables:
  MAVEN_OPTS: "-Dmaven.repo.local=${CI_PROJECT_DIR}/.m2/repository"
  MAVEN_CLI_OPTS: "--batch-mode --errors --fail-at-end --show-version"
  CACHE_KEY: ${CI_COMMIT_REF_SLUG}

cache:
  key: $CACHE_KEY
  paths:
    - .m2/repository


.build-module:
  stage: Build
  script:
    - echo "Building $MODULE"
    - mvn -pl $MODULE clean package --also-make
  artifacts:
    paths:
      - "*/target"
```

In the build job we pass the build template and the GitLab file that contains the variable content


```
60 # BUILD JOBS
61 build-clientui-module:
62   extends:
63     - .clientui-module
64     - .build-module
65
66 build-commande-module:
67   extends:
68     - .commande-module
69     - .build-module
70
71 build-paiement-module:
72   extends:
73     - .paiement-module
74     - .build-module
75
76 build-produit-module:
77   extends:
78     - .produit-module
79     - .build-module
```

The GitLab files for the different services are:


### Clientui Service:





Update manifests/commande.yaml, manifests/commande-config.yaml,...  
 Hamza Laouij authored 1 day ago







d5510382



Name	Last commit	Last update
..		
📁 .mvn/wrapper	Ajoute du squelette d'un client	3 years ago
📁 src	Update manifests/commande.yaml, manifests/commande-config.yaml, clientu...	1 day ago
🔥 .gitignore	Ajoute du squelette d'un client	3 years ago
🔥 .gitlab-ci.yml	Add new file	4 days ago
🐳 Dockerfile	Update Dockerfile	3 days ago
📄 mvnw	Ajoute du squelette d'un client	3 years ago
📄 mvnw.cmd	Ajoute du squelette d'un client	3 years ago
📄 pom.xml	Update pom.xml	3 days ago


 .gitlab-ci.yml
 
 97 Bytes

Edit in Web IDE
 
 Replace
 Delete
 




```

1  .clientui-module:
2  variables:
3    MODULE: "clientui"
4  only:
5    changes:
6      - "clientui/**/*"

```

## Commande Service:

ClientEtMSComm...
Devops / microservice-commandes /
History
Find file
Web IDE
Clone


Update .gitlab-ci.yml  
Hamza Laouij authored 2 days ago
da09fe21

Name	Last commit	Last update
..		
mvnw/wrapper	Ajoute du squelette d'un client	3 years ago
src	Update manifests/commande.yaml, manifests/commande-config.yaml, clientu...	1 day ago
.gitignore	Ajoute du squelette d'un client	3 years ago
.gitlab-ci.yml	Add new file	4 days ago
Dockerfile	Update Dockerfile	3 days ago
mvnw	Ajoute du squelette d'un client	3 years ago
mvnw.cmd	Ajoute du squelette d'un client	3 years ago
pom.xml	Update pom.xml	3 days ago

.gitlab-ci.yml 125 Bytes
Edit in Web IDE
Replace
Delete


```

1 .commande-module:
2   variables:
3     MODULE: "microservice-commandes"
4   only:
5     changes:
6       - "microservice-commandes/**/*"

```

## Paiment Service:

ClientEtMSComm...
Devops / microservice-paiement /
+
History
Find file
Web IDE
Download
Clone


Update manifests/commande.yaml, manifests/commande-config.yaml,...
Hamza Laouij authored 1 day ago
d5510382

Name	Last commit	Last update
..		
mvn/wrapper	Premier commit	3 years ago
src	Update manifests/commande.yaml, manifests/commande-config.yaml, clientu...	1 day ago
.gitignore	Premier commit	3 years ago
.gitlab-ci.yml	Update .gitlab-ci.yml file	18 hours ago
Dockerfile	Update Dockerfile	3 days ago
mvnw	Premier commit	3 years ago
mvnw.cmd	Premier commit	3 years ago
pom.xml	Update pom.xml	4 days ago

.gitlab-ci.yml 123 Bytes
Edit in Web IDE
Replace
Delete

```


1  .paiement-module:
2    variables:
3      MODULE: "microservice-paiement"
4    only:
5      changes:
6        - "microservice-paiement/**/*"

```

## Produit Service:

ClientEtMSComm... Devops / microservice-produits / +

History Find file Web IDE Clone


Delete VousEtesDansTestBranch.md
Hamza Laouij authored 21 seconds ago
14f56d4f

Name	Last commit	Last update
..		
└ .mvn/wrapper	Premier commit	3 years ago
└ src	Client fonctionnel communiquant avec tous les MS	3 years ago
└ .gitignore	Premier commit	3 years ago
└ .gitlab-ci.yml	Upload New File	4 days ago
└ Dockerfile	Update Dockerfile	3 days ago
└ mvnw	Premier commit	3 years ago
└ mvnw.cmd	Premier commit	3 years ago
└ pom.xml	Premier commit	3 years ago

.gitlab-ci.yml 122 Bytes Edit in Web IDE Replace Delete

1	.produit-module:
2	variables:
3	MODULE: "microservice-produits"
4	only:
5	changes:
6	- "microservice-produits/**/*"

Include files in the root pipeline:

14	include:
15	- local: clientui/.gitlab-ci.yml
16	- local: microservice-commandes/.gitlab-ci.yml
17	- local: microservice-paiement/.gitlab-ci.yml
18	- local: microservice-produits/.gitlab-ci.yml
19	- template: Security/SAST.gitlab-ci.yml
20	- template: Security/Container-Scanning.gitlab-ci.yml
21	- template: DAST.gitlab-ci.yml
22	

## Test Stage

Same as before, we will create test template and pass service name to it:

```
42  ∨ .test-module:
43      stage: Test
44  ∨  script:
45      - echo "Testing $MODULE"
46      - mvn $MAVEN_CLI_OPTS -pl $MODULE test --also-make
47
```

```
81  # TEST JOBS
82  ∨ test-clientui-module:
83      extends:
84      - .clientui-module
85      - .test-module
86
87  ∨ test-commande-module:
88      extends:
89      - .commande-module
90      - .test-module
91
92  ∨ test-paiement-module:
93  ∨  extends:
94      - .paiement-module
95      - .test-module
96
97  ∨ test-produit-module:
98  ∨  extends:
99      - .produit-module
100     - .test-module
```

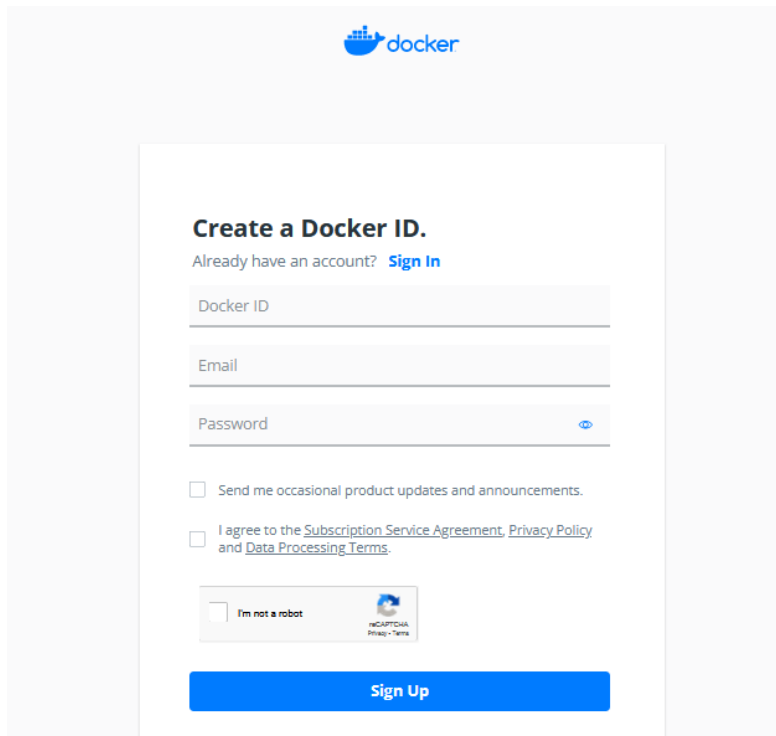


# Build And Push Docker Image

## Prerequisites:

- Docker Hub Account

## Create Docker Hub Account:

A screenshot of the Docker Hub sign-up page. At the top is the Docker logo. Below it, the heading 'Create a Docker ID.' is followed by the text 'Already have an account? [Sign In](#)'. The form contains three input fields: 'Docker ID', 'Email', and 'Password' (with an eye icon for toggling visibility). Below these fields are two checkboxes: 'Send me occasional product updates and announcements.' and 'I agree to the [Subscription Service Agreement](#), [Privacy Policy](#) and [Data Processing Terms](#)'. At the bottom of the form is a reCAPTCHA widget with the text 'I'm not a robot' and a 'Sign Up' button.

## Add credentials of Docker Hub to our pipeline variables:

**CI\_REGISTRY:** docker.io, the registry used to push docker images.

**CI\_REGISTRY\_PASS:** password of our Docker Hub account.

**CI\_REGISTRY\_USER:** username of our Docker Hub account.

The screenshot shows the GitLab CI/CD Settings page. On the left sidebar, the 'CI/CD' option is highlighted with a red arrow. The main content area is titled 'Variables' and contains a table of environment variables. A red arrow points to the 'CI\_REGISTRY' variable in the table.

**Artifacts**  
A job artifact is an archive of files and directories saved by a job when it finishes.

**Variables**  
Variables store information, like passwords and secret keys, that you can use in job scripts. [Learn more.](#)

Variables can be:

- Protected: Only exposed to protected branches or tags.
- Masked: Hidden in job logs. Must match masking requirements. [Learn more.](#)

Environment variables are configured by your administrator to be protected by default.

Type	Key	Value	Protected	Masked	Environments
Variable	CI_REGISTRY	*****	×	×	All (default)
Variable	CI_REGISTRY_PASS	*****	×	×	All (default)
Variable	CI_REGISTRY_USER	*****	×	×	All (default)

## Create Dockerfile for all the services:

The screenshot shows the GitLab repository view for the 'clientui' directory. The file 'Dockerfile' is selected, showing its content and metadata.

Hamza Laouij > Devops > Repository

ClientEtMScComm... Devops / clientui / **Dockerfile** Find file Blame History Permalink

Update Dockerfile  
Hamza Laouij authored 3 days ago 5769c170

Dockerfile 127 Bytes Edit in Web IDE Replace Delete

```

1 FROM maven:3.8.4-jdk-8
2 VOLUME /app
3 ADD target/*.jar clientui.jar
4 EXPOSE 8080 8080
5 ENTRYPOINT ["java", "-jar", "/clientui.jar"]

```

The screenshot shows the GitLab repository view for the 'microservice-commands' directory. The file 'Dockerfile' is selected, showing its content and metadata.

Hamza Laouij > Devops > Repository

ClientEtMScComm... Devops / microservice-commands / **Dockerfile** Find file Blame History Permalink

Update Dockerfile  
Hamza Laouij authored 3 days ago 122bc282

Dockerfile 130 Bytes Edit in Web IDE Replace Delete


```


1 FROM maven:3.8.4-jdk-8
2 VOLUME /app
3 ADD target/*.jar mcommandes.jar
4 EXPOSE 9002 9002
5 ENTRYPOINT ["java", "-jar", "mcommandes.jar"]

```

Hamza Laouij > Devops > Repository

ClientEtMSComm... Devops / microservice-paiement / **Dockerfile** Find file Blame History Permalink

 **Update Dockerfile**  
Hamza Laouij authored 3 days ago 52c535f1

 **Dockerfile** 129 Bytes Edit in Web IDE Replace Delete


```


1 FROM maven:3.8.4-jdk-8
2 VOLUME /app
3 COPY target/*.jar mpaielement.jar
4 EXPOSE 9003 9003
5 ENTRYPOINT ["java", "-jar", "mpaielement.jar"]

```

Hamza Laouij > Devops > Repository

ClientEtMSComm... Devops / microservice-produits / **Dockerfile** Find file Blame History Permalink

 **Update Dockerfile**  
Hamza Laouij authored 3 days ago 3a99dd34

 **Dockerfile** 128 Bytes Edit in Web IDE Replace Delete

```

1 FROM maven:3.8.4-jdk-8
2 VOLUME /app
3 ADD target/*.jar mproduits.jar
4 EXPOSE 9001 9001
5 ENTRYPOINT ["java", "-jar", "mproduits.jar"]

```

## Create the job template:

```

48 .push-image:
49   image: docker:latest
50   stage: Build and Push Docker Image
51   before_script:
52     - echo $CI_REGISTRY_PASS | docker login -u $CI_REGISTRY_USER $CI_REGISTRY --password-stdin
53   script:
54     - echo "Building and Pushing Docker Images"
55     - docker build -t h4mz4/$MODULE ./MODULE
56     - docker push h4mz4/$MODULE
57


















```

```

102 # Push Docker Image
103 <img alt="chevron down" data-bbox="165 103 175 113"/> push-clientui-image:
104 <img alt="chevron down" data-bbox="165 114 175 124"/> extends:
105     - .clientui-module
106     - .push-image
107
108 <img alt="chevron down" data-bbox="165 159 175 169"/> push-commande-image:
109 <img alt="chevron down" data-bbox="165 170 175 180"/> extends:
110     - .commande-module
111     - .push-image
112
113 <img alt="chevron down" data-bbox="165 215 175 225"/> push-paiement-image:
114 <img alt="chevron down" data-bbox="165 226 175 236"/> extends:
115     - .paiement-module
116     - .push-image
117 <img alt="chevron down" data-bbox="165 261 175 271"/> push-produit-image:
118 <img alt="chevron down" data-bbox="165 272 175 282"/> extends:
119     - .produit-module
120     - .push-image
121

```

## Final Result:

h4mz4	<img alt="chevron down" data-bbox="265 375 275 385"/>	 Search by repository name	<a href="#">Create Repository</a>
h4mz4 / <b>clientui</b> Updated 2 days ago	 Not Scanned	 0	 155  Public
h4mz4 / <b>microservice-paiement</b> Updated 2 days ago	 Not Scanned	 0	 68  Public
h4mz4 / <b>microservice-produits</b> Updated 2 days ago	 Not Scanned	 0	 63  Public
h4mz4 / <b>microservice-commandes</b> Updated 2 days ago	 Not Scanned	 0	 58  Public

# Deploy Stage:

## Create Deployment files:

Hamza Laouij > Devops > Repository

ClientEtMSComm... Devops / manifests / +

History Find file Web IDE Clone

Update manifests/clientui.yaml, manifests/commande.yaml, manifests/paiement.yaml, manifests/produit.yaml  
Hamza Laouij authored 1 day ago eb854d4f

Name	Last commit	Last update
..		
{ } clientui.yaml	Update manifests/clientui.yaml, manifests/commande.yaml, manifests/paiement.yaml, manifests/produit.yaml	1 day ago
{ } commande.yaml	Update manifests/clientui.yaml, manifests/commande.yaml, manifests/paiement.yaml, manifests/produit.yaml	1 day ago
{ } paiement.yaml	Update manifests/clientui.yaml, manifests/commande.yaml, manifests/paiement.yaml, manifests/produit.yaml	1 day ago
{ } produit.yaml	Update manifests/clientui.yaml, manifests/commande.yaml, manifests/paiement.yaml, manifests/produit.yaml	1 day ago

clientui.yaml 544 Bytes Edit in Web IDE Replace Delete

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: clientui-deployment
5    labels:
6      app: clientui
7  spec:
8    replicas: 1
9    selector:
10     matchLabels:
11       app: clientui
12    template:
13     metadata:
14       labels:
15         app: clientui
16     spec:
17       containers:
18         - name: clientui
19           image: h4mz4/clientui
20           ports:
21             - containerPort: 8080
22 ---
23 apiVersion: v1
24 kind: Service
25 metadata:
26   name: clientui
27 spec:
28   type: LoadBalancer
29   selector:
30     app: clientui
31   ports:
32     - protocol: TCP
33       port: 80
34       targetPort: 8080
```

commande.yaml 558 Bytes Edit in Web IDE Replace Delete

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: commande-deployment
5    labels:
6      app: commande
7  spec:
8    replicas: 1
9    selector:
10     matchLabels:
11       app: commande
12  template:
13    metadata:
14     labels:
15       app: commande
16    spec:
17     containers:
18     - name: commande
19       image: h4mz4/microservice-commandes
20       ports:
21       - containerPort: 9002
22  ---
23  apiVersion: v1
24  kind: Service
25  metadata:
26    name: commande
27  spec:
28    type: LoadBalancer
29    selector:
30      app: commande
31    ports:
32    - protocol: TCP
33      port: 80
34      targetPort: 9002
```

pairement.yaml557 Bytes

Edit in Web IDE

ReplaceDelete

```
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: paiement-deployment
5    labels:
6      app: paiement
7  spec:
8    replicas: 1
9    selector:
10     matchLabels:
11       app: paiement
12   template:
13     metadata:
14       labels:
15         app: paiement
16     spec:
17       containers:
18         - name: paiement
19           image: h4mz4/microservice-paiement
20           ports:
21             - containerPort: 9003
22   ---
23   apiVersion: v1
24   kind: Service
25   metadata:
26     name: paiement
27   spec:
28     type: LoadBalancer
29     selector:
30       app: paiement
31     ports:
32       - protocol: TCP
33         port: 80
34         targetPort: 9003
```

```
produit.yaml 550 Bytes Edit in Web IDE Replace Delete

1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: produit-deployment
5    labels:
6      app: produit
7  spec:
8    replicas: 1
9    selector:
10     matchLabels:
11       app: produit
12   template:
13     metadata:
14       labels:
15         app: produit
16     spec:
17       containers:
18         - name: produit
19           image: h4mz4/microservice-produits
20           ports:
21             - containerPort: 9001
22   ---
23   apiVersion: v1
24   kind: Service
25   metadata:
26     name: produit
27   spec:
28     type: LoadBalancer
29     selector:
30       app: produit
31     ports:
32       - protocol: TCP
33         port: 80
34         targetPort: 9001
```

## Create Google Cloud Account:

Google Kubernetes Engine (GKE) Contact Us Get started for free

Google Kubernetes Engine

Benefits

Key features

Customers

What's new

Documentation

Use cases

Continuous delivery pipeline

Migrating a two-tier

## Google Kubernetes Engine

A simple way to automatically deploy, scale, and manage Kubernetes.


New customers can use \$300 in free credits to try out GKE.

Try GKE free Contact sales

✓ Run Kubernetes on a platform built by the largest engineering contributor to K8s

✓ Start quickly with single-click clusters and scale up to 15000 nodes

✓ Leverage a high-availability control plane including multi-zonal and regional clusters



VIDEO

### What is Google Kubernetes Engine (GKE)?

6:30

## Create GKE Project:



Select a project

NEW PROJECT

Search projects and folders

RECENT

STARRED

ALL

	Name	ID
✓ ☆	Devops ?	devops-340313
☆	My First Project ?	rugged-filament-340313

CANCEL

OPEN

## Create GKE Cluster:

We choose GKE Standard:

Create cluster

Select the cluster mode that you want to use.

Compare cluster modes to learn more about their differences. [COMPARE](#)

**GKE Standard**  
A pay-per-node Kubernetes cluster where you configure and manage your nodes.  
[Learn more](#)

CONFIGURE

**GKE Autopilot**  
A pay-per-Pod Kubernetes cluster where GKE manages your nodes with minimal configuration required. [Learn more](#)

CONFIGURE

CANCEL

← Create a Kubernetes cluster ADD NODE POOL REMOVE NODE POOL USE A SETUP GUIDE

- Cluster basics

**Cluster basics**

The new cluster will be created with the name, version, and in the location you specify here. After the cluster is created, name and location can't be changed.

**NAME** To experiment with an affordable cluster, try **My first cluster** in the **Cluster set-up guides**

Name

Location type

☒ Zonal

☐ Regional

Zone

**CREATE** **CANCEL** Equivalent **REST** or **COMMAND LINE**

Google Cloud Platform Devops Search Products, resources, docs (/)

Kubernetes Engine Kubernetes clusters CREATE DEPLOY REFRESH OPERATIONS

**OVERVIEW** **COST OPTIMIZATION**

Filter Enter property name or value

Status	Name	Location	Number of nodes	Total vCPUs	Total memory	Notifications	Labels
<input checked="" type="checkbox"/>	cluster	us-central1-c	3	6	12 GB	—	⋮

## Create Service Account:

Service Account gives permissions to GitLab to deploy on GKE cluster

**IAM & Admin** Service accounts + CREATE SERVICE ACCOUNT DELETE MANAGE ACCESS

**Service accounts for project "Devops"**

A service account represents a Google Cloud service identity, such as code running on Compute Engine VMs, App Engine apps, or systems running outside Google. [Learn more about service accounts.](#)

Organization policies can be used to secure service accounts and block risky service account features, such as automatic IAM Grants, key creation/upload, or the creation of service accounts entirely. [Learn more about service account organization policies.](#)

Filter Enter property name or value

Email	Status	Name	Description	Key ID	Actions
<input checked="" type="checkbox"/> 334933722393-compute@developer.gserviceaccount.com	<input checked="" type="checkbox"/>	Compute Engine default service account		No keys	⋮

## Create service account

### 1 Service account details

**Service account name \***  
devops

Display name for this service account

**Service account ID \***  
devops @devops-340313.iam.gserviceaccount.com ✕ ↺

**Service account description**  
Describe what this service account will do

CREATE AND CONTINUE

### 2 Grant this service account access to project (optional)

We give it the necessary roles:

#### Service account permissions (optional)

Grant this service account access to pv-lb-test so that it has permission to complete specific actions on the resources in your project. [Learn more](#)

<b>Role</b> Kubernetes Engine Develo... ▼	✕
Full access to Kubernetes API objects inside Kubernetes Clusters.	
<b>Role</b> Cloud Build Service Accou... ▼	✕
Can perform builds	
<b>Role</b> Storage Admin ▼	✕
Full control of GCS resources.	
<b>Role</b> Viewer ▼	✕
Read access to all resources.	

### 3 Grant users access to this service account (optional)

Grant access to users or groups that need to perform actions as this service account. [Learn more](#)

Service account users role ?

Grant users the permissions to deploy jobs and VMs with this service account

Service account admins role ?

Grant users the permission to administer this service account

DONE

CANCEL

## Create JSON Key:

DETAILS PERMISSIONS KEYS METRICS LOGS

Keys

Service account keys could pose a security risk if compromised. We recommend you avoid downloading service account keys and instead use the [Workload Identity Federation](#). You can learn more about the best way to authenticate service accounts on Google Cloud [here](#).

Add a new key pair or upload a public key certificate from an existing key pair.

Block service account key creation using [organization policies](#).  
[Learn more about setting organization policies for service accounts](#)

ADD KEY

Create new key

Upload existing key

Key	Key creation date	Key expiration date	
128e4c14a010c3be508886b5c58a64667724e6ea	Feb 6, 2022	Jan 1, 10000	

## Create private key for "gitlab"

Downloads a file that contains the private key. Store the file securely because this key can't be recovered if lost.

### Key type

☒ JSON

Recommended













☐ P12


For backward compatibility with code using the P12 format

CANCEL

CREATE

## Add JSON Key to Pipeline Variables:

Type	↑ Key	Value	Protected	Masked	Environments	
Variable	CI_REGISTRY 	***** 	×	×	All (default)	
Variable	CI_REGISTRY_PASS 	***** 	×	×	All (default)	
Variable	CI_REGISTRY_USER 	***** 	×	×	All (default)	
Variable	SERVICE_ACCOUNT_KEY 	***** 	✓	×	All (default)	



Add variable Reveal values

## Create Deploy Job:

```
146 # Deploy
147 deploy:
148   stage: Deploy
149   image: google/cloud-sdk
150   script:
151     - echo "$SERVICE_ACCOUNT_KEY" > key.json
152     - gcloud auth activate-service-account --key-file=key.json
153     - gcloud config set project devops-340313
154     - gcloud config set container/cluster cluster
155     - gcloud config set compute/zone us-central1-c
156     - gcloud container clusters get-credentials cluster --zone us-central1-c --project devops-340313
157     - kubectl apply -f manifests/produit.yaml
158     - kubectl apply -f manifests/commande.yaml
159     - kubectl apply -f manifests/paiement.yaml
160     - kubectl apply -f manifests/clientui.yaml
161
```

# Security Tests

Include security tests templates:

```
14   include:
15     - local: clientui/.gitlab-ci.yml
16     - local: microservice-commandes/.gitlab-ci.yml
17     - local: microservice-paiement/.gitlab-ci.yml
18     - local: microservice-produits/.gitlab-ci.yml
19     - template: Security/SAST.gitlab-ci.yml
20     - template: Security/Container-Scanning.gitlab-ci.yml
21     - template: DAST.gitlab-ci.yml
22
```

## SAST:

SAST is automatically running by adding its template

## Container Scanning:

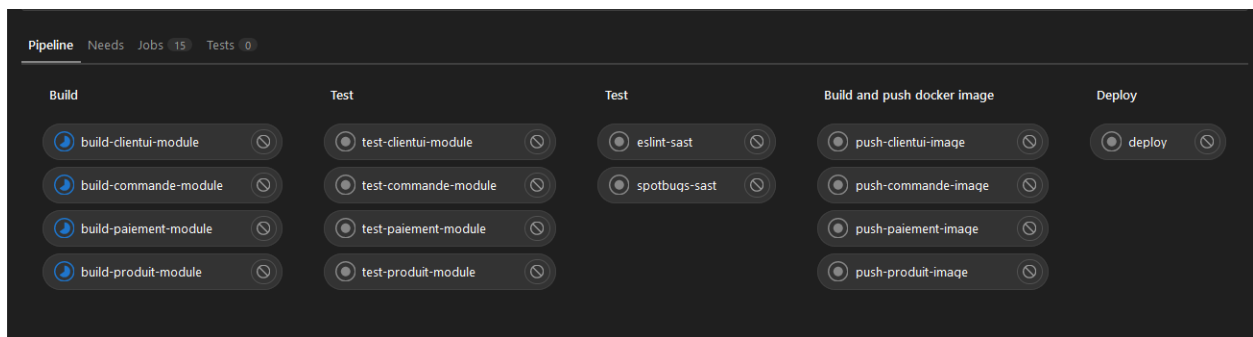
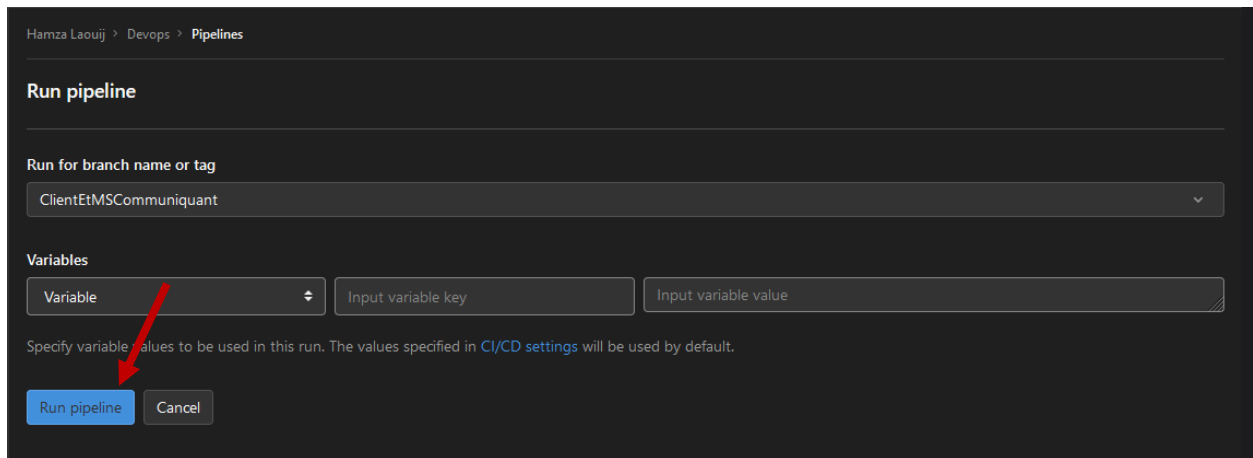
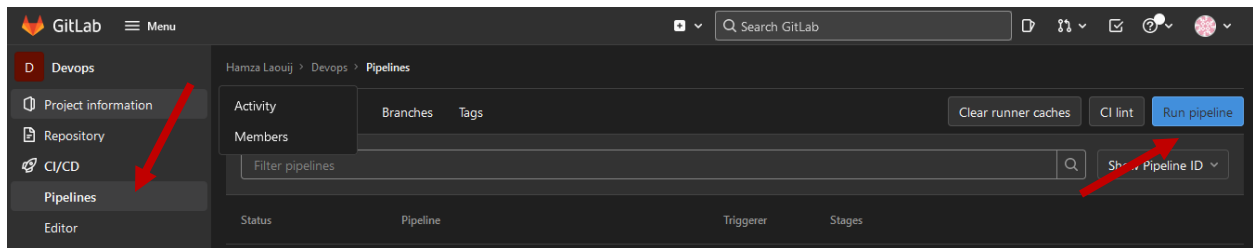
```
122  # Container Scanning
123  container_scanning:
124    stage: test
125    variables:
126      DOCKER_IMAGE: h4mz4/clientui
127
128  commande_scanning:
129    extends:
130      - container_scanning
131    variables:
132      DOCKER_IMAGE: h4mz4/microservice-commandes
133
134  paiement_scanning:
135    extends:
136      - container_scanning
137    variables:
138      DOCKER_IMAGE: h4mz4/microservice-paiement
139
140  produit_scanning:
141    extends:
142      - container_scanning
143    variables:
144      DOCKER_IMAGE: h4mz4/microservice-produits
```

## DAST:

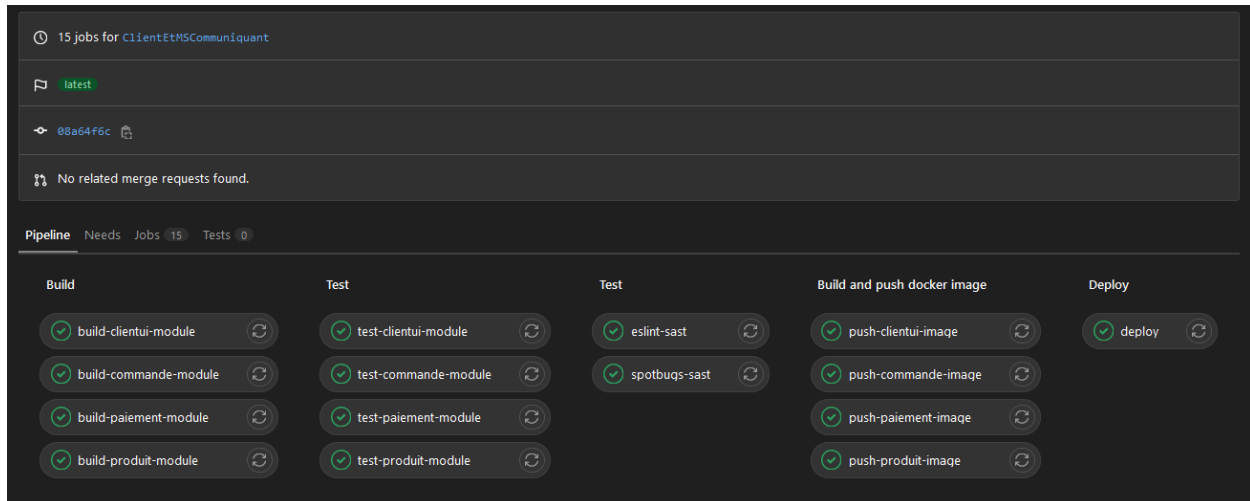
In DAST, we have to pass the website to the DAST\_WEBSITE variable

```
162 # DAST
163 dast:
164   variables:
165     DAST_WEBSITE: http://35.232.13.225/
166
```

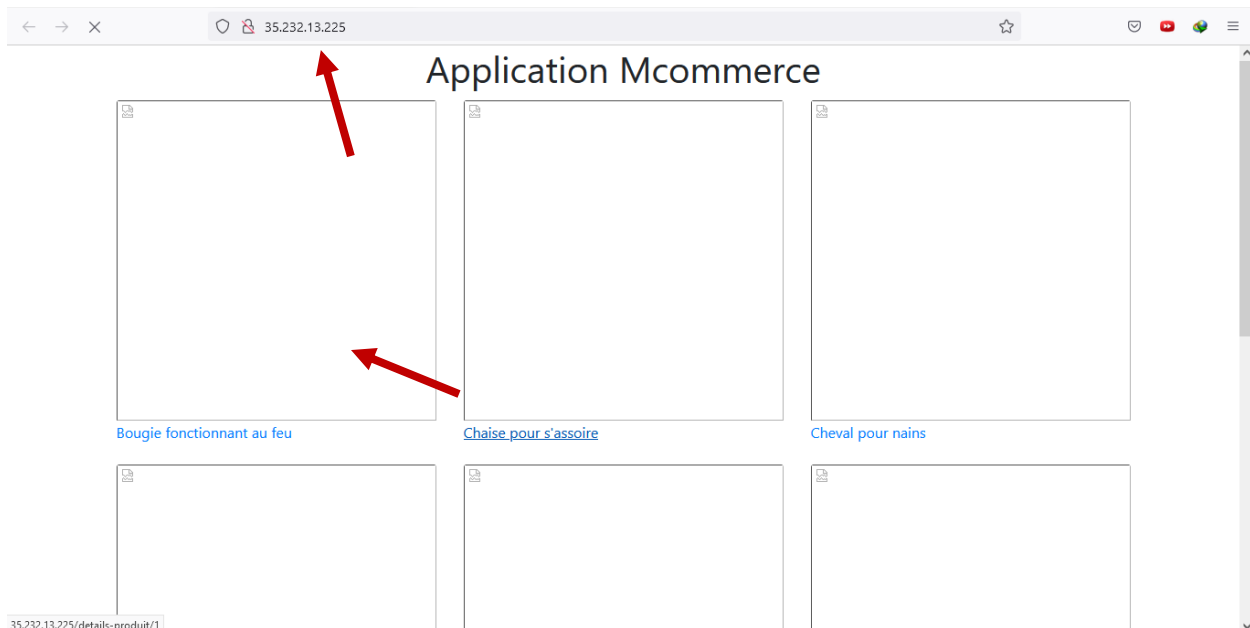
# Let's Run The Pipeline !







All jobs succeeded and the website is deployed on the url:  
<http://35.232.13.225/>



Let's try to click on "Bougie fonctionnant au feu"



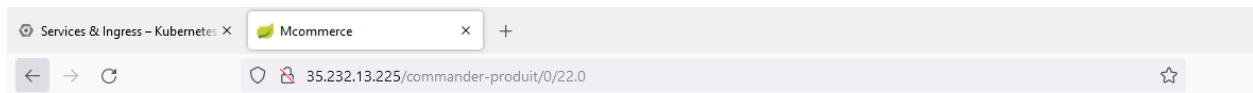
## Application Mcommerce



Bougie fonctionnant au feu

bougie qui fonctionne comme une ampoule mais sans électricité !

[COMMANDER](#)



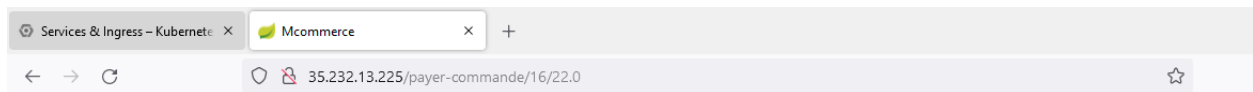
## Application Mcommerce

Ici l'utilisateur sélectionne en temps normal un moyen de paiement et entre les informations de sa carte bancaire.

Nous allons éviter d'ajouter les formulaires nécessaires afin de garder l'application la plus basique et simple possible pour la suite.

Si vous vous sentez à l'aise, vous pouvez créer un formulaire pour accepter le numéro de la CB, que vous traiterez dans le contrôleur grâce à un *PostMapping*.

[Payer Ma Commande](#)



## Application Mcommerce

Paieement Accepté

Everything works fine !

Let's get a closer look at every job:

**BUILD:**

225 [INFO] Building jar: /builds/LjHamza/Devops/clientui/target/clientui-0.0.1-SNAPSHOT.jar

226 [INFO]

227 [INFO] --- spring-boot-maven-plugin:2.0.0.RELEASE:repackage (default) @ clientui ---

228 [INFO] -----

229 [INFO] Reactor Summary:

230 [INFO]

231 [INFO] spring-boot-starter-parent 2.0.0.RELEASE ..... SUCCESS [ 0.260 s]

232 [INFO] clientui 0.0.1-SNAPSHOT ..... SUCCESS [ 9.884 s]

233 [INFO] -----

234 [INFO] BUILD SUCCESS

235 [INFO] -----

236 [INFO] Total time: 10.822 s

237 [INFO] Finished at: 2022-02-07T18:08:20Z

238 [INFO] -----

240 Saving cache for successful job 00:05

241 Creating cache clientetmscommuniquant-6...

242 .m2/repository: found 3911 matching files and directories

243 Uploading cache.zip to https://storage.googleapis.com/gitlab-com-runners-cache/project/33390385/clientetmscommuniquant-6

244 Created cache

246 Uploading artifacts for successful job 00:13

247 Uploading artifacts...

248 .m2/: found 3912 matching files and directories

249 \*/target: found 43 matching files and directories

250 Uploading artifacts as "archive" to coordinator... 201 Created id=2064008247 responseStatus=201 Created token=JyPLGzFs

252 Cleaning up project directory and file based variables 00:01

254 Job succeeded

build-clientui-mo... Retry

Duration: 1 minute 48 seconds

Finished: 38 minutes ago

Timeout: 1h (from project) ⓘ

Runner: #12270845 (JLgUopmM)

1-green.shared.runners-

manager.gitlab.com/default

Job artifacts

These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.

Keep Download Browse

Commit 08a64f6c ⓘ

Update microservice-produts/src/main/resources/data.sql

✓ Pipeline #465373957 for ClientEtmsCommuniquant ⓘ

Build

→ build-clientui-module

## TEST:

228 [INFO] -----

229 [INFO] T E S T S

230 [INFO] -----

231 [INFO]

232 [INFO] Results:

233 [INFO]

234 [INFO] Tests run: 0, Failures: 0, Errors: 0, Skipped: 0

235 [INFO]

236 [INFO] -----

237 [INFO] Reactor Summary:

238 [INFO]

239 [INFO] spring-boot-starter-parent 2.0.0.RELEASE ..... SUCCESS [ 0.008 s]

240 [INFO] clientui 0.0.1-SNAPSHOT ..... SUCCESS [ 8.965 s]

241 [INFO] -----

242 [INFO] BUILD SUCCESS

243 [INFO] -----

244 [INFO] Total time: 9.684 s

245 [INFO] Finished at: 2022-02-07T18:10:31Z

246 [INFO] -----

248 Saving cache for successful job 00:05

249 Creating cache clientetmscommuniquant-6...

250 .m2/repository: found 3911 matching files and directories

251 Uploading cache.zip to https://storage.googleapis.com/gitlab-com-runners-cache/project/33390385/clientetmscommuniquant-6

252 Created cache

254 Cleaning up project directory and file based variables 00:01

256 Job succeeded

test-clientui-mod... Retry

Duration: 1 minute 48 seconds

Finished: 37 minutes ago

Timeout: 1h (from project) ⓘ

Runner: #12270845 (JLgUopmM)

1-green.shared.runners-

manager.gitlab.com/default

Commit 08a64f6c ⓘ

Update microservice-produts/src/main/resources/data.sql

✓ Pipeline #465373957 for ClientEtmsCommuniquant ⓘ

Test

→ test-clientui-module

test-commande-module

test-paiement-module

test-produit-module

## Security TEST:

118 Downloading artifacts for build-paiement-module (2064008252)...

119 Downloading artifacts from coordinator... ok id=2064008252 responseStatus=200 OK token=4MdgB5y4

120 Downloading artifacts for build-produit-module (2064008255)...

121 Downloading artifacts from coordinator... ok id=2064008255 responseStatus=200 OK token=Yz8upi3R

123 Executing "step\_script" stage of the job script 00:02

124 Using docker image sha256:61102171e31cb1418b6d5f416fd21a180efe7cf9913a528725ff8c114da436a0 for registry.gitlab.com/gitlab-org/security-products/analyzers/eslint:2 with digest registry.gitlab.com/gitlab-org/security-products/analyzers/eslint@sha256:5c6b734db98056c375bf212e328d861e853ca2479abf725f2a989277b758bc3b ...

125 \$ /analyzer run

126 [INFO] [ESLint] [2022-02-07T18:12:06Z] ▶ Gitlab ESLint analyzer v2.24.3

127 [INFO] [ESLint] [2022-02-07T18:12:06Z] ▶ Detecting project

128 [INFO] [ESLint] [2022-02-07T18:12:06Z] ▶ Found project in /builds/ljhamza/Devops/clientui/target/classes/templates

129 [INFO] [ESLint] [2022-02-07T18:12:06Z] ▶ Running analyzer

130 [INFO] [ESLint] [2022-02-07T18:12:06Z] ▶ Making paths relative

131 [INFO] [ESLint] [2022-02-07T18:12:06Z] ▶ Creating report

133 Saving cache for successful job 00:05

134 Creating cache clientetmscommuniquant-6...

135 .m2/repository: found 3911 matching files and directories

136 Uploading cache.zip to https://storage.googleapis.com/gitlab-com-runners-cache/project/33390385/clientetmscommuniquant-6

137 Created cache

139 Uploading artifacts for successful job 00:01

140 Uploading artifacts...

141 gl-sast-report.json: found 1 matching files and directories

142 Uploading artifacts as "sast" to coordinator... 201 Created id=2064008265 responseStatus=201 Created token=ozDsZj3C

144 Cleaning up project directory and file based variables 00:01

146 Job succeeded

eslint-sast

Retry

Duration: 1 minute 28 seconds

Finished: 36 minutes ago

Timeout: 1h (from project) ?

Runner: #12270845 (JLgUopmM)  
1-green.shared.runners-manager.gitlab.com/default

Job artifacts  
These artifacts are the latest. They will not be deleted (even if expired) until newer artifacts are available.

Commit 08a646c  
Update microservice-produits/src/main/resources/data.sql

Pipeline #465373957 for ClientEtmsCommuniquant

test

→ eslint-sast

spotbugs-sast

## Build And Push Docker Images:

213 0b0f2f2f5279: Preparing

214 6be21046fcff: Waiting

215 1e5f4c3d671c: Waiting

216 613ab28cf833: Waiting

217 bed676ceab7a: Waiting

218 6398d5cccd2c: Waiting

219 0b0f2f2f5279: Waiting

220 f90b1621d190b: Layer already exists

221 8f3a1bd592e3: Layer already exists

222 f53ed9ea8a10: Layer already exists

223 cc0471e51515: Layer already exists

224 6be21046fcff: Layer already exists

225 1e5f4c3d671c: Layer already exists

226 bed676ceab7a: Layer already exists

227 613ab28cf833: Layer already exists

228 6398d5cccd2c: Layer already exists

229 0b0f2f2f5279: Layer already exists

230 e7c4f2cddb9c: Pushed

231 latest: digest: sha256:50dae9ddd3740ac4324a0b6cce42058a426b61801adb8e3189a9cde47eaad8b31 size: 2632

233 Saving cache for successful job 00:06

234 Creating cache clientetmscommuniquant-6...

235 .m2/repository: found 3911 matching files and directories

236 Uploading cache.zip to https://storage.googleapis.com/gitlab-com-runners-cache/project/33390385/clientetmscommuniquant-6

237 Created cache

239 Cleaning up project directory and file based variables 00:00

241 Job succeeded

push-clientui-image

Retry

Duration: 1 minute 43 seconds

Finished: 31 minutes ago

Timeout: 1h (from project) ?

Runner: #12270845 (JLgUopmM)  
1-green.shared.runners-manager.gitlab.com/default

Commit 08a646c  
Update microservice-produits/src/main/resources/data.sql

Pipeline #465373957 for ClientEtmsCommuniquant

Build and Push Docker Image

→ push-clientui-image

push-commande-image

push-paiement-image

push-produit-image

Let's see Docker Hub:

h4mz4

▼

🔍

Search by repository name

Create Repository

h4mz4 / **microservice-paiement**

Updated 33 minutes ago

🛡️ Not Scanned

☆ 0

📶 70

🔓 Public

h4mz4 / **microservice-produits**

Updated 33 minutes ago

🛡️ Not Scanned

☆ 0

📶 65

🔓 Public

h4mz4 / **clientui**

Updated 33 minutes ago

🛡️ Not Scanned

☆ 0

📶 157

🔓 Public

h4mz4 / **microservice-commandes**

Updated 33 minutes ago

🛡️ Not Scanned

☆ 0

📶 60

🔓 Public

## Deploy Job:

```

129 $ gcloud config set container/cluster cluster
130 Updated property [container/cluster].
131 $ gcloud config set compute/zone us-central1-c
132 Updated property [compute/zone].
133 $ gcloud container clusters get-credentials cluster --zone us-central1-c --project devops-340313
134 Fetching cluster endpoint and auth data.
135 kubeconfig entry generated for cluster.
136 $ kubectl apply -f manifests/produit.yaml
137 deployment.apps/produit-deployment unchanged
138 service/produit unchanged
139 $ kubectl apply -f manifests/commande.yaml
140 deployment.apps/commande-deployment unchanged
141 service/commande unchanged
142 $ kubectl apply -f manifests/paiement.yaml
143 deployment.apps/paiement-deployment unchanged
144 service/paiement unchanged
145 $ kubectl apply -f manifests/clientui.yaml
146 deployment.apps/clientui-deployment unchanged
147 service/clientui unchanged
149 Saving cache for successful job
150 Creating cache clientetmscommunicant-6...
151 .m2/repository: found 3911 matching files and directories
152 Uploading cache.zip to https://storage.googleapis.com/gitlab-com-runners-cache/project/33390385/clientetmscommunicant-6
153 Created cache
155 Cleaning up project directory and file based variables
157 Job succeeded

```

deploy

Duration: 3 minutes 0 seconds

Finished: 31 minutes ago

Timeout: 1h (from project)

Runner: #12270859 (xS6Vzpvo)

5-green.shared.runners-manager.gitlab.com/default

Commit 08a64f6c

Update microservice-produits/src/main/resources/data.sql

Pipeline #465373957 for ClientEtMSCommunicant

Deploy

→ deploy

## GKE Deployments:

Google Cloud Platform

DevOps

Search Products, resources, docs (/)

Kubernetes Engine

Workloads

REFRESH

DEPLOY

DELETE

OPERATIONS

Cluster

Namespace

RESET

SAVE

Workloads are deployable units of computing that can be created and managed in a cluster.

OVERVIEW

COST OPTIMIZATION

Filter

Is system object : False

Filter workloads

X

?

|||

<input type="checkbox"/>	Name ↑	Status	Type	Pods	Namespace	Cluster
<input type="checkbox"/>	clientui-deployment	OK	Deployment	1/1	default	cluster
<input type="checkbox"/>	commande-deployment	OK	Deployment	1/1	default	cluster
<input type="checkbox"/>	paiement-deployment	OK	Deployment	1/1	default	cluster
<input type="checkbox"/>	produit-deployment	OK	Deployment	1/1	default	cluster

## GKE Services:

Google Cloud Platform

DevOps

Search Products, resources, docs (/)

Kubernetes Engine

Services & Ingress

REFRESH

CREATE INGRESS

DELETE

Cluster

Namespace

RESET

SAVE

SERVICES

INGRESS

Services are sets of Pods with a network endpoint that can be used for discovery and load balancing. Ingresses are collections of rules for routing external HTTP(S) traffic to Services.

Filter

Is system object : False

Filter services and ingresses

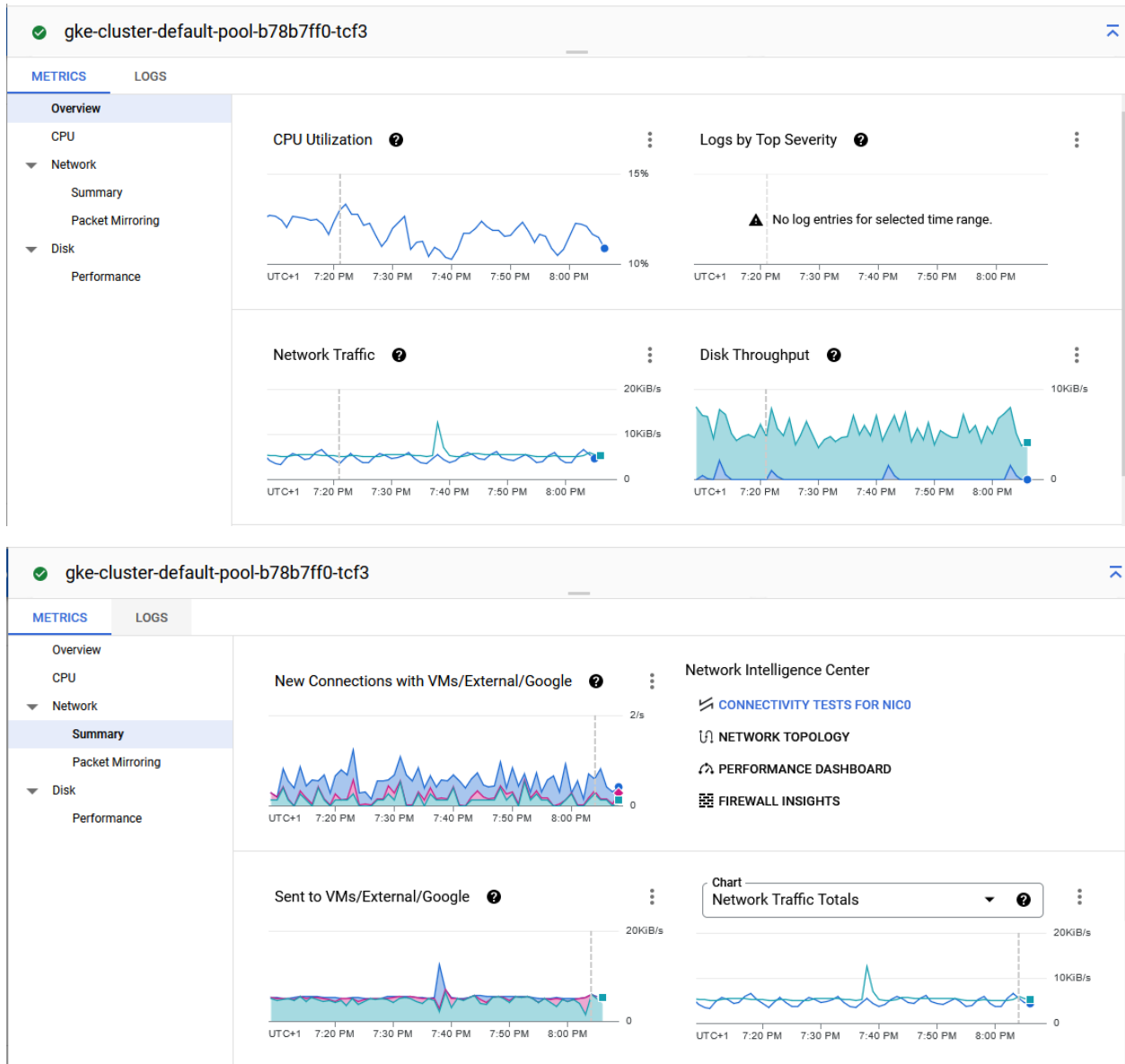
X

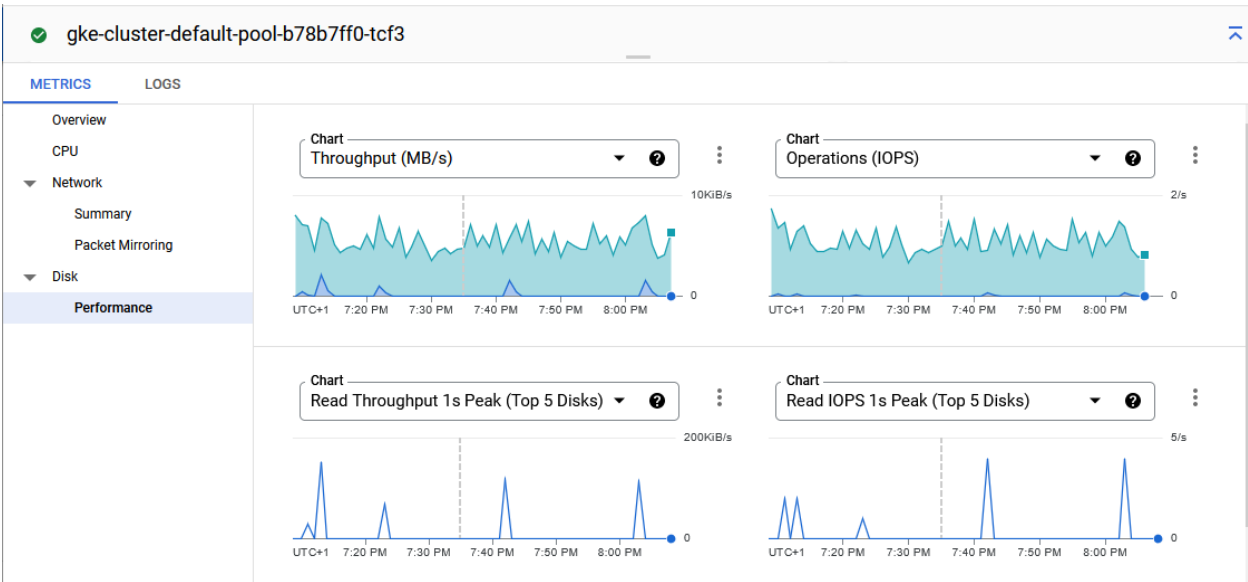
?

|||

<input type="checkbox"/>	Name ↑	Status	Type	Endpoints	Pods	Namespace	Clusters
<input type="checkbox"/>	clientui	OK	External load balancer	35.232.13.225:80	1/1	default	cluster
<input type="checkbox"/>	commande	OK	External load balancer	34.122.181.26:80	1/1	default	cluster
<input type="checkbox"/>	paiement	OK	External load balancer	34.121.90.99:80	1/1	default	cluster
<input type="checkbox"/>	produit	OK	External load balancer	34.135.96.155:80	1/1	default	cluster

# Monitoring







## Project Resources

**GitLab Project:** <https://gitlab.com/LjHamza/Devops/>

**Docker Hub:** <https://hub.docker.com/u/h4mz4>

**APP Website:** <http://35.232.13.225/>