

République Tunisienne

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

Université de Monastir

Institut Supérieur d'Informatique et de Mathématiques de Monastir



Rapport de projet Devops

Deuxième Année Cycle Ingénieur en Informatique

Spécialité :

Génie Logiciel

Elaboré par :

Khawla Grissa
Malak Letaief
Samar Mestiri

Déploiement d'une application web avec pipeline CI/CD en local

Encadré par : Mr. Lazhar HAMEL
Mme. Dorra DHAOU

Année Universitaire : 2024/2025

Objectif du Projet

L'objectif principal de ce projet est de créer une application web simple, de la gérer avec un système de versioning (Git), de configurer une pipeline CI/CD pour l'automatisation des tests et de la mise à jour du code, puis de déployer l'application localement à l'aide de Docker.

Ce projet se concentre sur les aspects de configuration, intégration et déploiement, et non sur le développement fonctionnel de l'application.

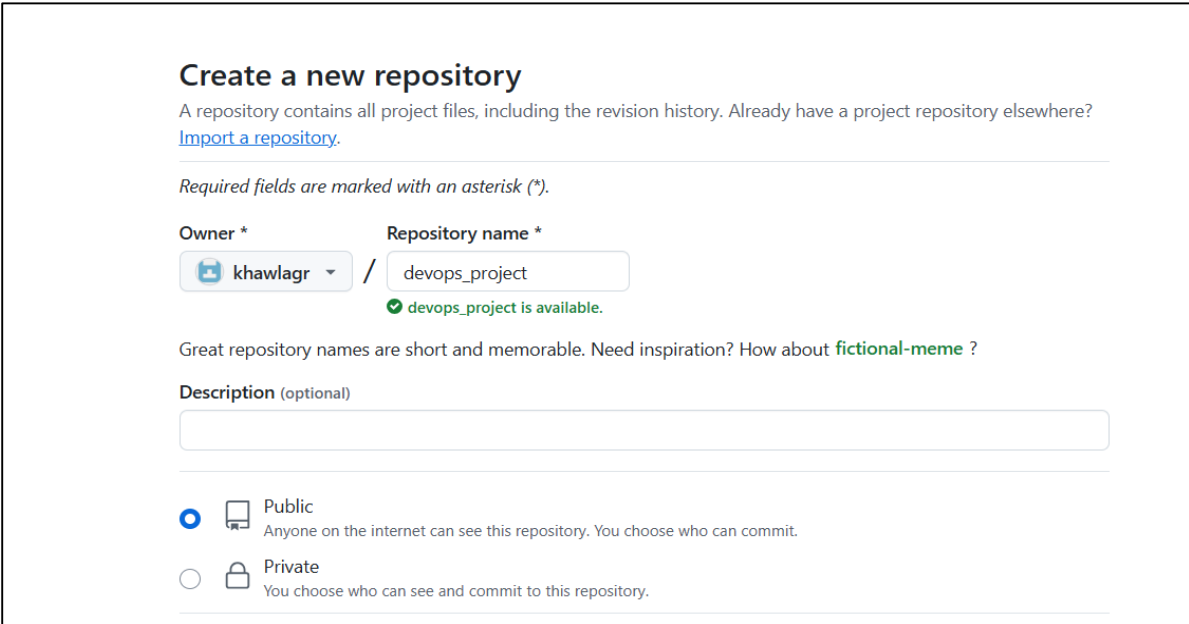
Outils et Environnement Utilisés

- **Express.js** (framework backend Node.js)
- **HTML/CSS, EJS** (frontend)
- **MongoDB** (base de données NoSQL)
- **Git et GitHub** (versionnage)
- **Docker** (conteneurisation)
- **GitHub Actions** (CI/CD)
- **Terraform** pour l'infrastructure-as-code

Étapes Réalisées

1. Initialisation du Version Control

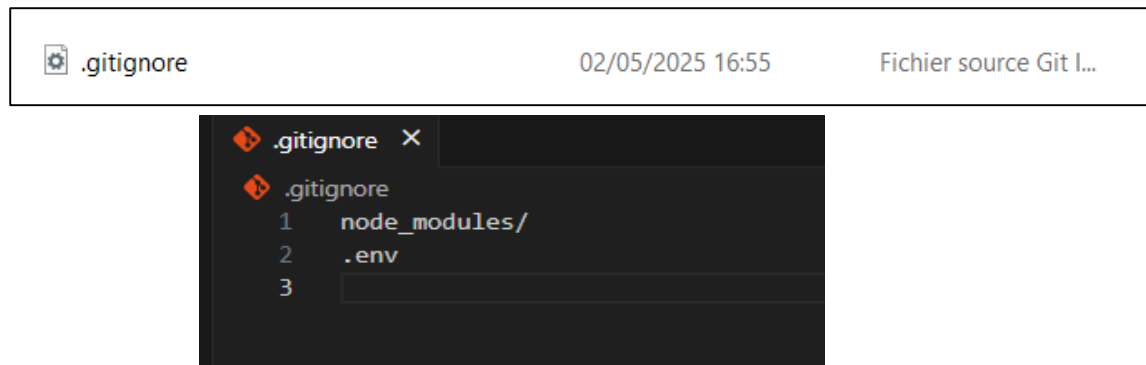
Création d'un dépôt GitHub nommé `devops-project`.



The screenshot shows the GitHub 'Create a new repository' page. At the top, it says 'Create a new repository' and provides a brief explanation of what a repository is. Below this, there is a link to 'Import a repository'. A note states 'Required fields are marked with an asterisk (*)'. The form has two main sections: 'Owner' and 'Repository name'. The 'Owner' field is a dropdown menu showing 'khawlagr'. The 'Repository name' field is a text input containing 'devops_project'. Below the name field, a green checkmark indicates 'devops_project is available.'. There is a suggestion for repository names: 'Great repository names are short and memorable. Need inspiration? How about fictional-meme?'. Below this is a 'Description (optional)' text area. At the bottom, there are two radio buttons for visibility: 'Public' (selected) and 'Private'. The 'Public' option is described as 'Anyone on the internet can see this repository. You choose who can commit.' The 'Private' option is described as 'You choose who can see and commit to this repository.'

Clonage local : `git clone` <https://github.com/khawlagr/devops-project.git>

Ajout d'un fichier **.gitignore**



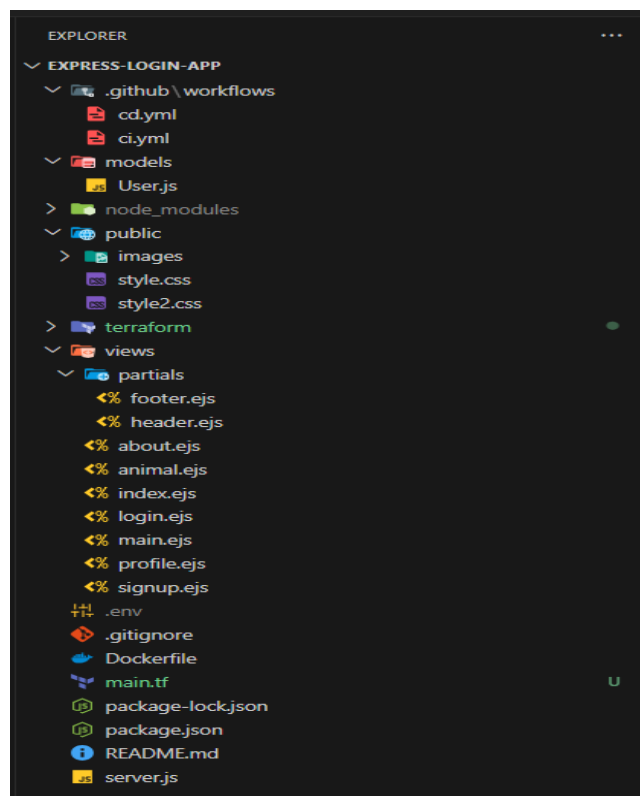
Commit et push initial :

```
git add .
git commit -m "Initial commit"
git push origin main
```

2. Développement de l'Application

Structure 3 couches :

- **Model** : Connexion MongoDB
- **View** : Template EJS
- **Controller** : Express.js



index.ejs

```
server.js  index.ejs X
views > index.ejs > ?
1  <%- include('./partials/header') %>
2
3  <section class="hero">
4    <div class="hero-content">
5      <h1>Discover the Amazing World of Animals</h1>
6      <p>Learn about different species from around the globe</p>
7    </div>
8  </section>
9
10 <section class="animal-grid">
11   <h2>Featured Animals</h2>
12
13   <div class="animals">
14     <% animals.forEach(animal => { %>
15       <div class="animal-card">
16         ">
17         <div class="animal-info">
18           <h3><%= animal.name %></h3>
19           <p class="species"><%= animal.species %></p>
20           <p><%= animal.description.substring(0, 100) %>...</p>
21           <a href="/animal/<%= animal.id %>" class="btn">Learn More</a>
22         </div>
23       </div>
24     <% }); %>
25   </div>
26 </section>
27
28 <%- include('./partials/footer') %>
```

Server.js

```
server.js X
server.js > ...
1  require('dotenv').config();
2  const express = require('express');
3  const mongoose = require('mongoose');
4  const session = require('express-session');
5  const bcrypt = require('bcrypt');
6  const User = require('./models/User');
7  const path=require('path');
8
9  const app = express();
10
11 app.set('view engine', 'ejs');
12 app.use(express.static('public'));
13 app.use(express.urlencoded({ extended: true }));
14
15 app.use(session({
16   secret: process.env.SESSION_SECRET,
17   resave: false,
18   saveUninitialized: true
19 }));
20
21 mongoose.connect(process.env.MONGO_URI)
22   .then(() => console.log("MongoDB connected"))
23   .catch(err => console.error(err));
24
25
26 // Accueil
27 app.get('/', (req, res) => {
28   res.render('main', { user: req.session.user });
29 });
30
31
```

Installer **nodemon** :

```
PS C:\express-login-app> npm install --save-dev nodemon
>>

up to date, audited 293 packages in 2s

51 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS C:\express-login-app> 
```

Modifier le fichier **package.json** :

```
package.json X
package.json > {} devDependencies
1  {
2    "name": "express-login-app",
3    "version": "1.0.0",
4    "main": "server.js",
5    "scripts": {
6      "dev": "nodemon server.js"
7    },
8    "dependencies": {
9      "bcrypt": "^5.1.0",
10     "dotenv": "^16.0.3",
11     "ejs": "^3.1.9",
12     "express": "^4.18.2",
13     "express-session": "^1.17.3",
14     "mongodb": "^6.16.0",
15     "mongoose": "^7.2.0"
16   },
17   "devDependencies": {
18     "mocha": "^11.2.2",
19     "nodemon": "^3.1.10"
20   }
21 }
22
```

Lancer le projet en mode développement :

Accéder à : <http://localhost:3000>

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

PS C:\express-login-app> npm run dev

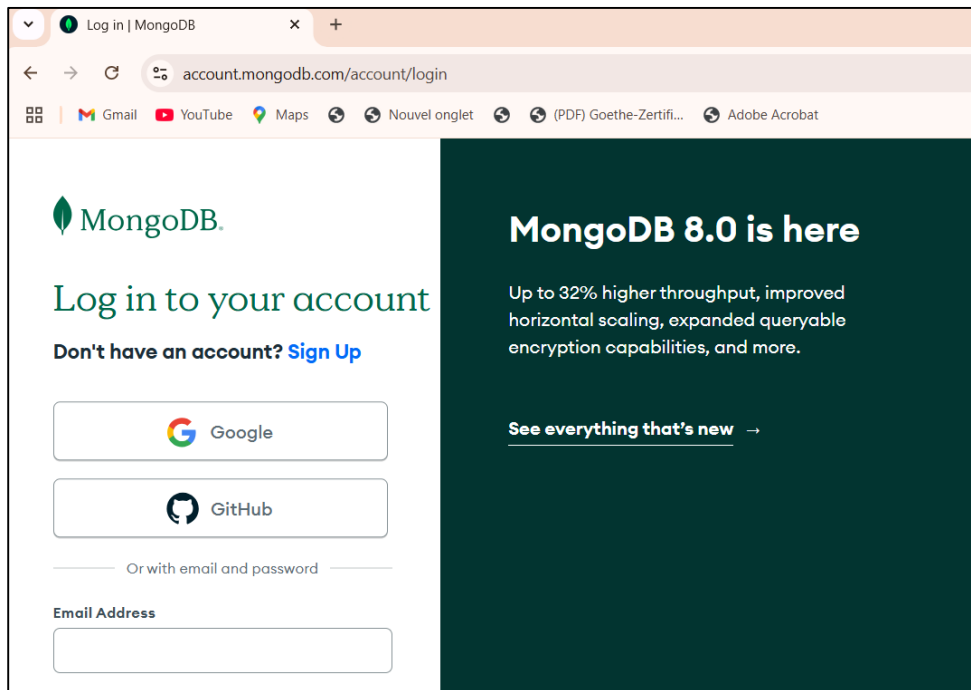
> express-login-app@1.0.0 dev
> nodemon server.js

[nodemon] 3.1.10
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,cjs,json
[nodemon] starting `node server.js`
(node:18756) [DEP0040] DeprecationWarning: The `punycode` module is deprecated. Please use a userland alternative instead.
(Use `node --trace-deprecation ...` to show where the warning was created)
http://localhost:3000
MongoDB connected

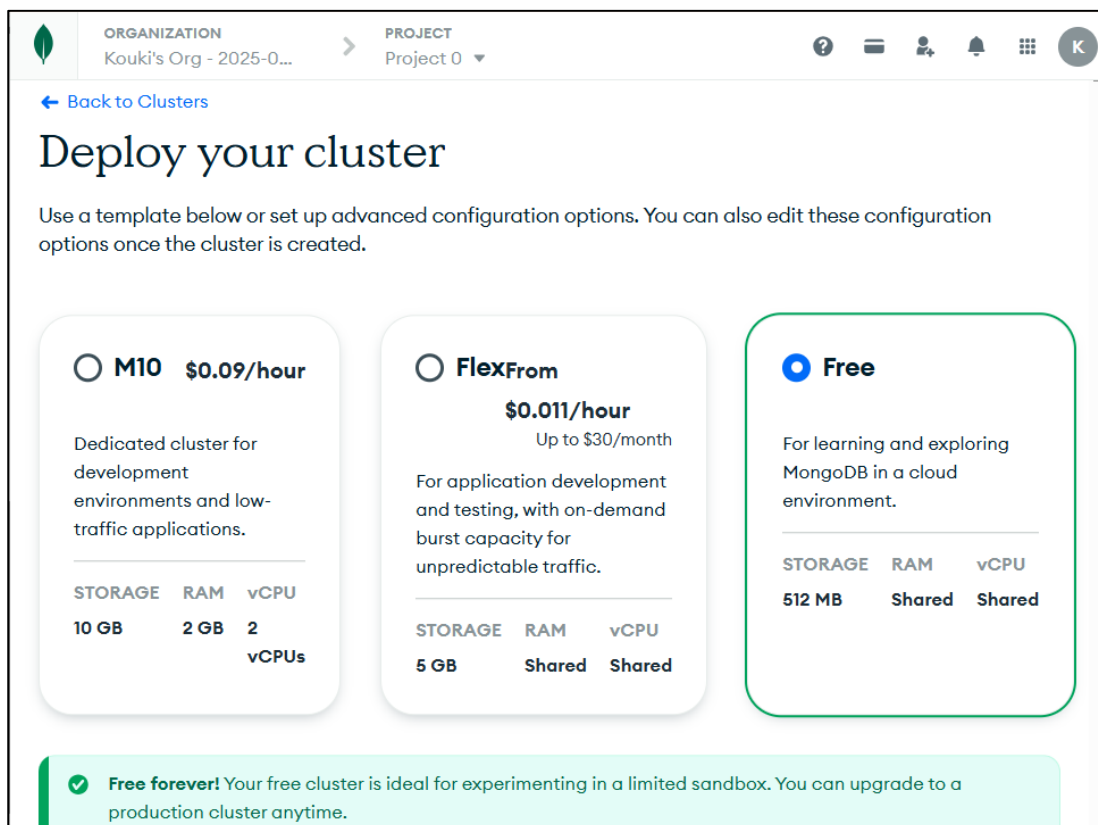
```

Installation MongoDB

Créer un compte sur <https://www.mongodb.com/cloud/atlas>



Créer un cluster gratuit






Sélectionner le fournisseur cloud : **AWS**

Nommer le cluster : **Cluster0**


Configurations


Name
You cannot change the name once the cluster is created.

Provider



Quick setup

☐ Automate security setup 

☐ Preload sample dataset 

Créer un utilisateur pour la base

Connect to Cluster0

1

2

3

Set up connection security

Choose a connection method

Connect


You need to secure your MongoDB Atlas cluster before you can use it. Set which users and IP addresses can access your cluster now. [Read more](#)

- Add a connection IP address**

✓ Your current IP address (197.238.236.212) has been added to enable local connectivity. Only an IP address you add to your Access List will be able to connect to your project's clusters. Add more later in [Network Access](#).
- Create a database user**

This first user will have [atlasAdmin](#) permissions for this project.


We autogenerated a username and password. You can use this or create your own.

 You'll need your database user's credentials in the next step. Copy the database user password.

Username

Password

HIDE

 **Copy**

Création d'une base de données **devops_db**

Create Database

Database name ?

devops_db

Collection name ?

users

Additional Preferences

Select

Cancel

Create

Overview

DATABASE

Clusters

SERVICES

Atlas Search

Stream Processing

Triggers

Migration

Data Federation

SECURITY

Quickstart

Backup

KHAWLA'S ORG - 2025-04-25 > PROJECT 0 > DATABASES

ClusterO

Overview

Real Time

Metrics

Collections

Atlas Search

Query Insights

Performance Advisor

DATABASES: 2 COLLECTIONS: 7

+ Create Database

Search Namespaces

devops_db

users

sample_mflix

devops_db.users

STORAGE SIZE: 34KB LOGICAL DATA SIZE: 981B TOTAL DOCUMENTS: 7 INDEXES TOTAL SIZE: 72KB

Find

Indexes

Schema Anti-Patterns

Aggregation

Search Indexes

Generate queries from natural language in Compass

Filter

Type a query: { field: 'value' }

Application :


Page d'accueil

Bienvenue sur notre site


Connexion | Inscription


Page d'inscription

[Retour à l'accueil](#)



Créer un compte






Déjà inscrit ? [Connexion](#)


Création d'un nouveau compte


Étude de cas : Création d'un nouveau compte Asma

[Retour à l'accueil](#)



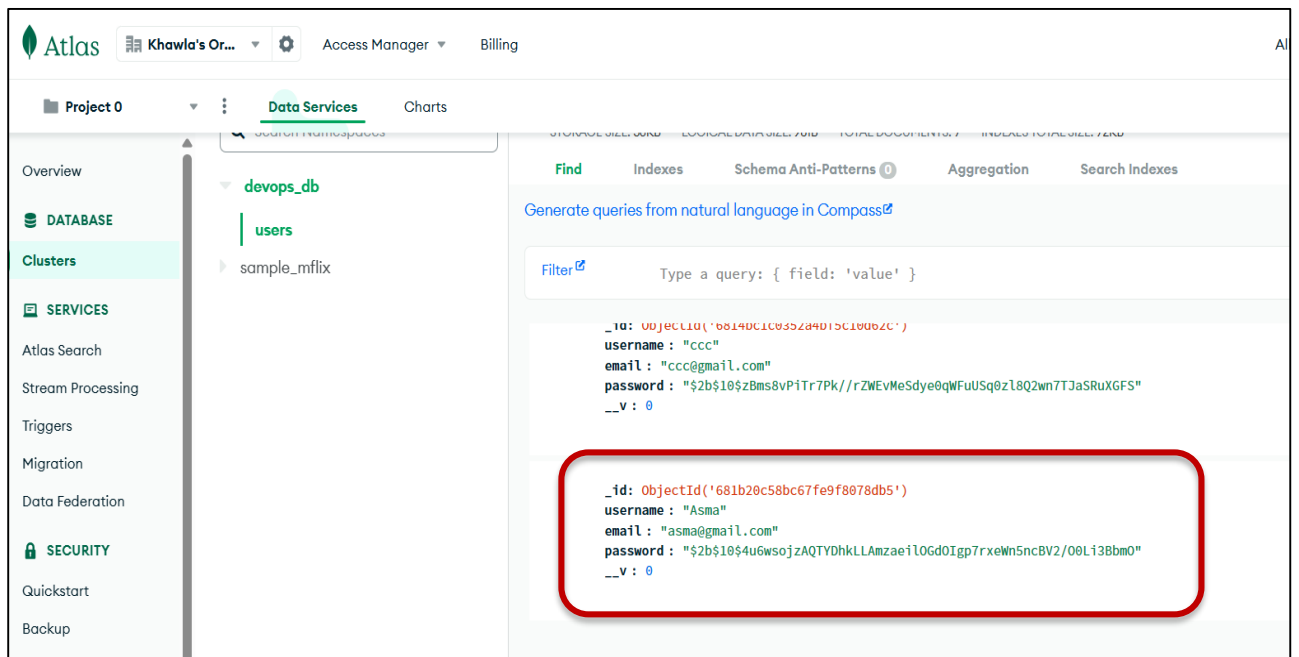
Créer un compte





Déjà inscrit ? [Connexion](#)

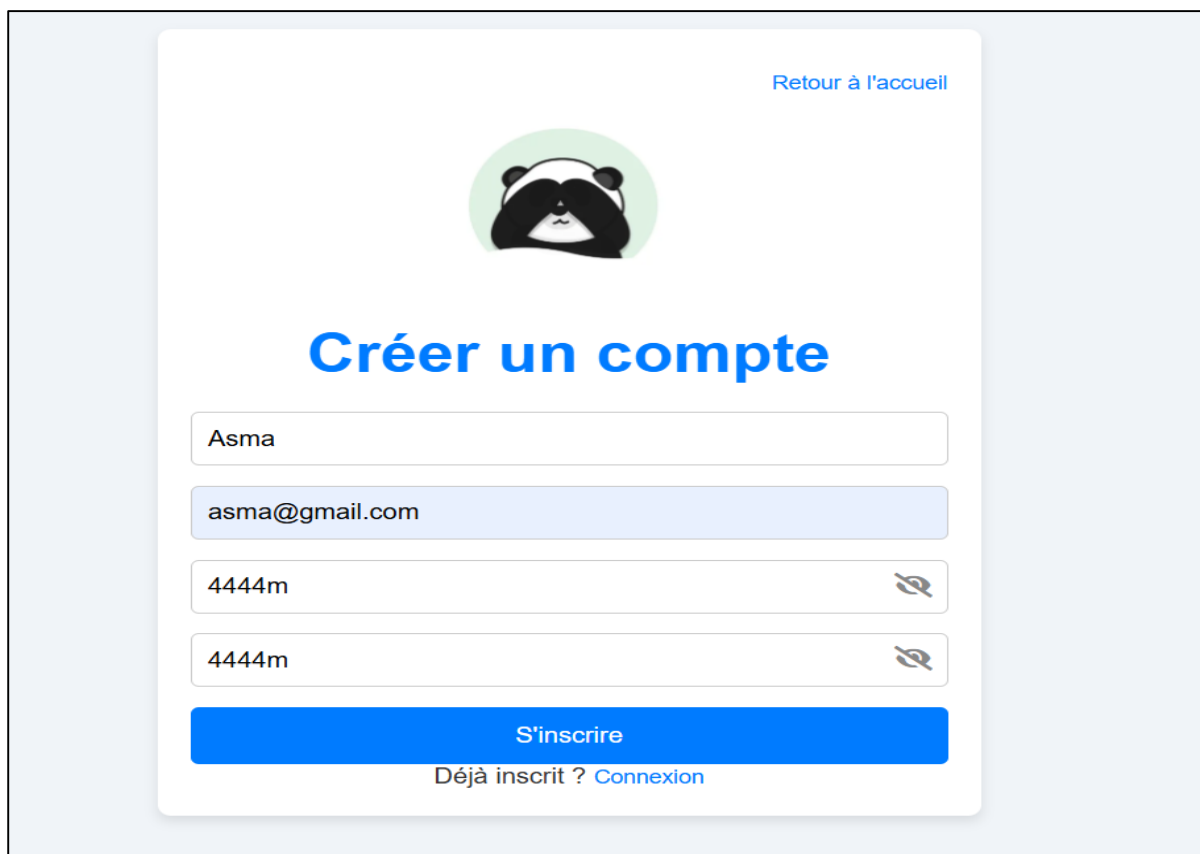
Utilisateur ajouté à la base de données



The screenshot shows the Atlas Data Services interface. On the left, a sidebar contains navigation links: Overview, DATABASE, Clusters, SERVICES, Atlas Search, Stream Processing, Triggers, Migration, Data Federation, SECURITY, Quickstart, and Backup. The main panel displays the 'devops_db' database with a 'users' collection. A query filter is applied, showing two user records. The second record is highlighted with a red box:

```
{
  "_id": ObjectId('681b20c58bc67fe9f8078db5'),
  "username": "Asma",
  "email": "asma@gmail.com",
  "password": "$2b$10$4u6wsojzAQTYDhKLLAmzaeiLOgd0Igp7rxWn5ncBV2/00Li3Bbm0",
  "__v": 0
}
```

Un compte avec cette adresse e-mail existe déjà.



The registration form is titled "Créer un compte" and features a panda logo. It includes a "Retour à l'accueil" link at the top right. The form fields are:

- Username: Asma
- Email: asma@gmail.com
- Password: 4444m
- Confirm Password: 4444m

Below the fields is a blue "S'inscrire" button. At the bottom, there is a link "Déjà inscrit ? Connexion".

[Retour à l'accueil](#)



Créer un compte

Nom d'utilisateur ou email déjà utilisé



S'inscrire

Déjà inscrit ? [Connexion](#)

Page de connexion

[Retour à l'accueil](#)



Login




Login

Don't have an account? [Sign Up](#)

Cas 1 : Adresse e-mail invalide. Veuillez entrer une adresse valide.

[Retour à l'accueil](#)



Login

!


Veuillez inclure "@" dans l'adresse e-mail. Il manque un symbole "@" dans "Asma".

Login

Don't have an account? [Sign Up](#)

Cas 2 : Mot de passe incorrect. Veuillez réessayer

[Retour à l'accueil](#)



Login


Mot de passe incorrect

Login

Don't have an account? [Sign Up](#)

Cas 3 : Utilisateur introuvable. Veuillez vérifier vos informations de connexion.


[Retour à l'accueil](#)



Login

Utilisateur introuvable

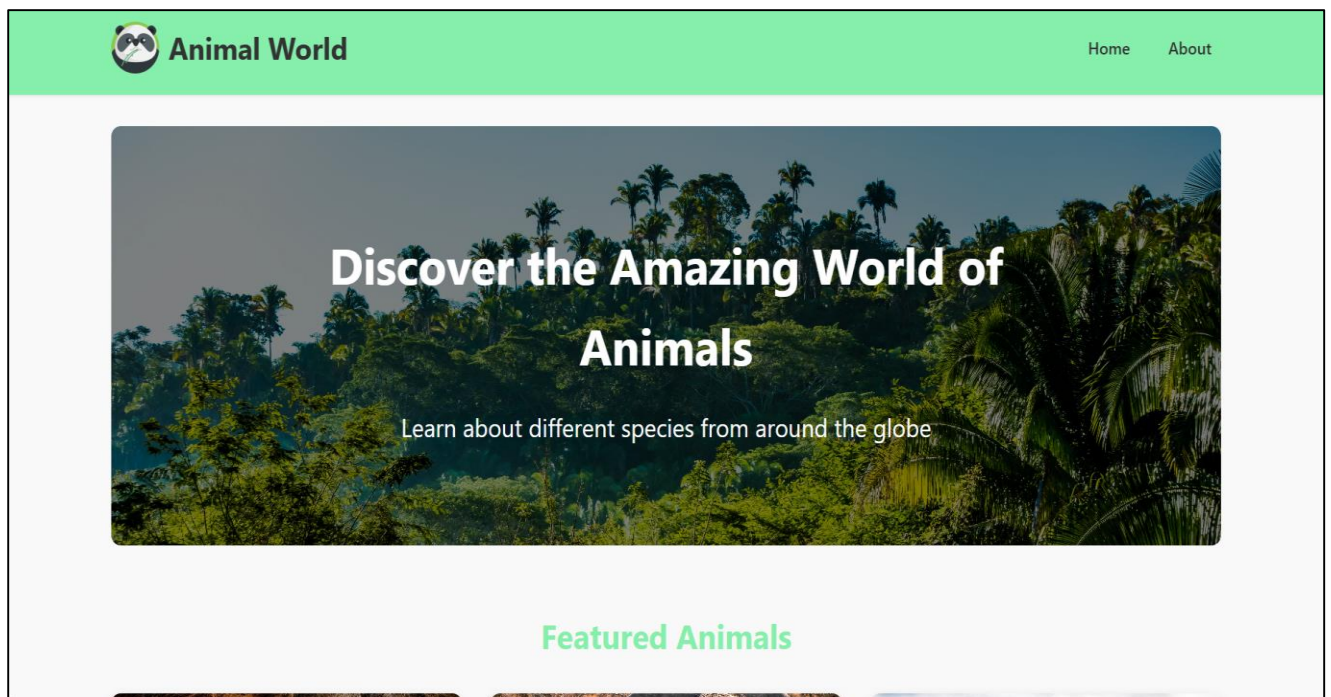
asma22@gmail.com

..... 

Login

Don't have an account? [Sign Up](#)

Cas 4 : Compte ouvert avec succès. Bienvenue dans votre espace "**Animals World**".




Le concept de notre site web est de permettre aux utilisateurs de **découvrir différents animaux** à travers une interface simple et interactive.

Le site propose plusieurs fonctionnalités, notamment :


- Un bouton **"Learn more"** pour en savoir plus sur chaque animal.
- Deux boutons de navigation en haut de la page : **"About"** (pour en apprendre davantage sur le site) et **"Home"** (pour revenir à la page d'accueil).

Featured Animals




Lion
Panthera leo
The lion is a large cat of the genus Panthera native to Africa and India. It has a muscular, broad-c...

[Learn More](#)




African Elephant
Loxodonta africana
African elephants are the largest existing land animals. They have a trunk, tusks, large ear flaps, ...

[Learn More](#)




Emperor Penguin
Aptenodytes forsteri
Emperor penguins are the tallest and heaviest of all living penguin species and are endemic to Antar...

[Learn More](#)




Bengal Tiger
Panthera tigris tigris
The Bengal tiger is a tiger subspecies native to the Indian subcontinent. It is recognizable by its ...

[Learn More](#)




Giant Panda
Ailuropoda melanoleuca
The giant panda is a bear species native to China. It is characterized by its bold black and white c...

[Learn More](#)




Peregrine Falcon
Falco peregrinus
The peregrine falcon is a widespread bird of prey known for its incredible speed during dives, makin...

[Learn More](#)




Monarch Butterfly
Danaus plexippus
The monarch butterfly is known for its incredible multi-generational migration from North America to...

[Learn More](#)



Bald Eagle
Haliaeetus leucocephalus
The bald eagle is a bird of prey found in North America and is the national bird of the United State...

[Learn More](#)



Poison Dart Frog
Dendrobatidae
Poison dart frogs are known for their bright coloration and the potent toxins secreted through their...

[Learn More](#)



Great White Shark

Carcharodon carcharias

The great white shark is a large predatory shark known for its size and powerful jaws. It is an apex...

[Learn More](#)



Octopus

Octopoda

Octopuses are highly intelligent marine mollusks characterized by their eight arms and soft bodies. ...

[Learn More](#)



Green Sea Turtle

Chelonia mydas

Green sea turtles are large sea turtles known for their herbivorous diet as adults. They are an impo...

[Learn More](#)



Cheetah

Acinonyx jubatus

The cheetah is a large cat native to Africa and central Iran. It is the fastest land animal, capable...

[Learn More](#)



Humpback Whale

Megaptera novaeangliae

Humpback whales are baleen whales known for their distinctive body shape, long pectoral fins, and ac...

[Learn More](#)



Snow Leopard

Panthera uncia

The snow leopard is a large cat native to the mountain ranges of Central and South Asia. They have t...

[Learn More](#)



Macaw

Ara (various species)

Macaws are long-tailed, often colorful New World parrots. They are known for their vibrant plumage, ...

[Learn More](#)



Komodo Dragon

Varanus komodoensis

The Komodo dragon is the largest living lizard species, endemic to several Indonesian islands. They ...

[Learn More](#)



Sea Otter


Enhydra lutris

Sea otters are marine mammals native to the coasts of the northern and eastern North Pacific Ocean. ...

[Learn More](#)

Lion

Panthera leo



Classification

Category: Mammal

Species: Panthera leo

Habitat


Savanna, grasslands

About

The lion is a large cat of the genus Panthera native to Africa and India. It has a muscular, broad-chested body, short, rounded head, round ears, and a hairy tuft at the end of its tail. Known for their social structure called prides.


Back to All Animals

© 2025 Animal World. All rights reserved.

Animal World

HomeAbout

About Animal World



Our Mission

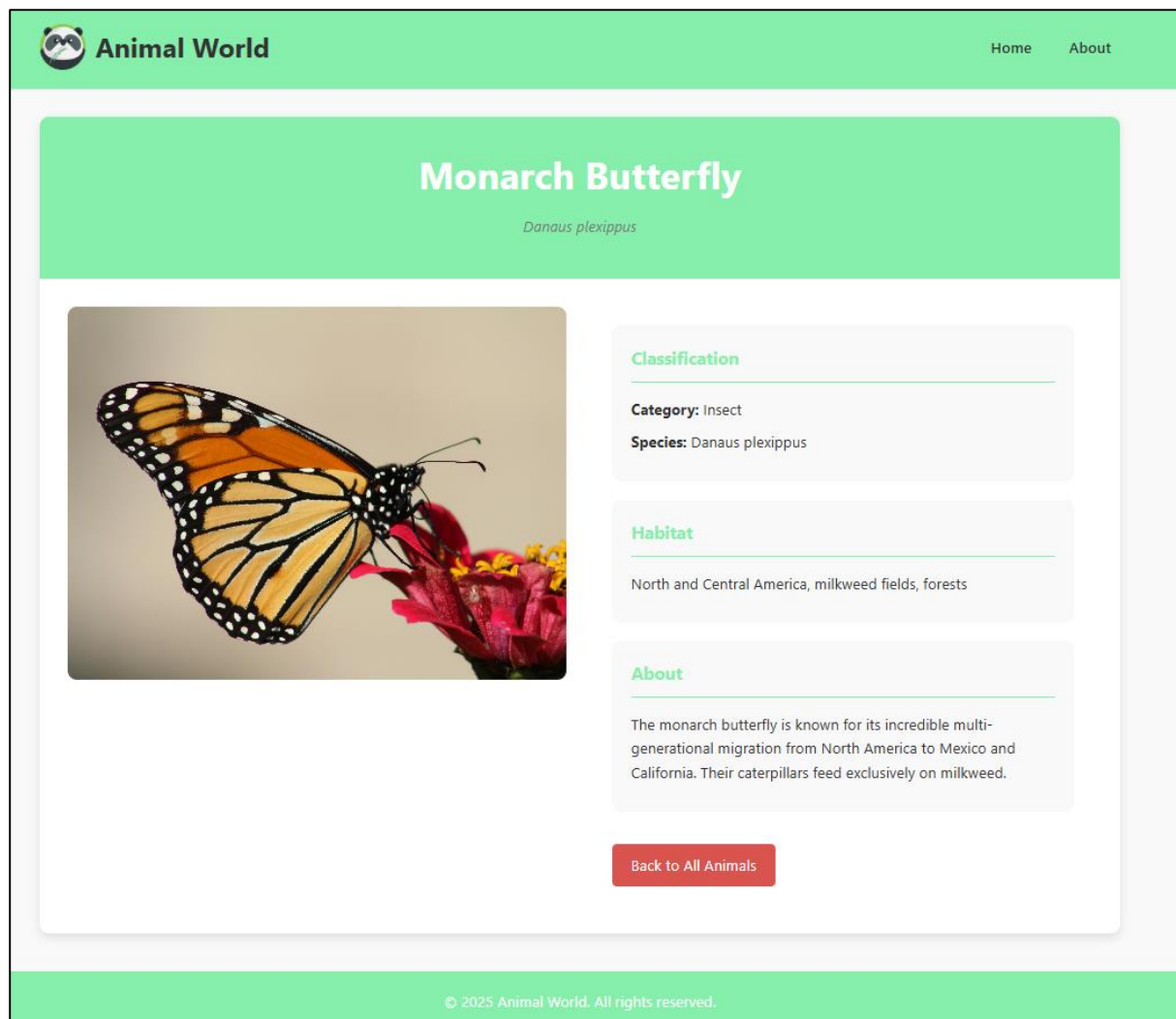
Animal World is dedicated to educating people about the diverse wildlife that inhabits our planet. Our mission is to promote understanding and appreciation of animals from all ecosystems, fostering a sense of responsibility toward wildlife conservation.

What We Do

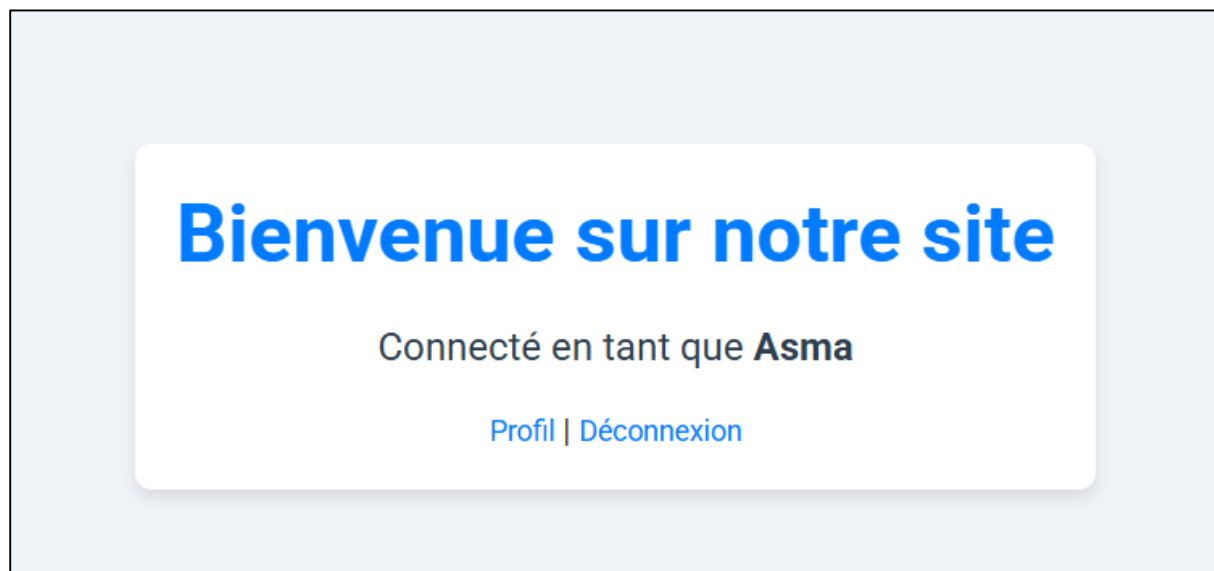
We collect and present information about various animal species, their habitats, behaviors, and conservation status. Our team of experts works diligently to ensure that all information is accurate, up-to-date, and presented in an engaging manner.

Join Us

Whether you're a wildlife enthusiast, a student, or simply curious about the animal kingdom, Animal World provides valuable resources to expand your knowledge. Explore our collection of animal profiles and discover the wonders of wildlife.



Un lien "**Déconnexion**" est disponible pour permettre à l'utilisateur de quitter sa session. Lorsqu'il clique sur ce lien, il est redirigé vers la page de connexion ou la page d'accueil publique, selon la configuration du site.



3. Containeriser l'application

Préparer un Dockerfile pour Express

```
Dockerfile

Fichier  Modifier  Affichage

FROM node:14

RUN apt-get update && apt-get install -y \
    build-essential \
    python3 \
    && rm -rf /var/lib/apt/lists/*

WORKDIR /app

COPY package*.json ./

RUN npm install

COPY . .

EXPOSE 3000

CMD ["node", "server.js"]
```

Construire l'image Docker : **docker build -t myapp .**

```
C:\express-login-app>docker build -t myapp .
[*] Building 855.6s (10/10) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 174B
=> [internal] load metadata for docker.io/library/node:14
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/5] FROM docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461a 783.5s
=> => resolve docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461a 0.2s
=> => sha256:0d27a8e861329007574c6766fba946d48e20d2c8e964e873de352603f22c4ceb 450B / 450B 0.3s
=> => sha256:0c8cc2f24a4dc6b4e602e086fc9446b0a541e8acd9ad72d2e90df3ba22f158b3 2.29MB / 2.29MB 19.5s
=> => sha256:6f51ee005deac0d99898e41b8ce60ebf250ebe1a31a0b03f613aec6bbcb9b3d8 4.19kB / 4.19kB 0.9s
=> => sha256:5f32ed3c3f278edda4fc571c880b5277355a29ae8f52b52cdf865f058378a590 35.24MB / 35.24MB 324.8s
=> => sha256:d9a8df5894511ce28a05e2925a75e8a4acbd0634c39ad734fdffa8e23d1b1569 191.85MB / 191.85MB 765.7s
=> => sha256:1de76e268b103d05fa8960e0f77951ff54b012b63429c34f5d6adfd09f5f9ee2 51.88MB / 51.88MB 397.6s
=> => sha256:3d2201bd995cccfc12851a50820de03d34a17011dcb9ac9fd3a50c952cbb131 10.00MB / 10.00MB 84.8s
=> => sha256:b253aeafeaa7e0671bb60008df01de101a38a045ff7bc656e3b0fbfc7c05cca5 7.80MB / 7.80MB 81.9s
=> => sha256:2ff1d7c41c74a25258bfa6f0b8adb0a727f84518f55f65ca845ebc747976c408 50.45MB / 50.45MB 313.0s
=> => extracting sha256:2ff1d7c41c74a25258bfa6f0b8adb0a727f84518f55f65ca845ebc747976c408 2.8s
=> => extracting sha256:b253aeafeaa7e0671bb60008df01de101a38a045ff7bc656e3b0fbfc7c05cca5 0.4s
=> => extracting sha256:3d2201bd995cccfc12851a50820de03d34a17011dcb9ac9fd3a50c952cbb131 0.3s
=> => extracting sha256:1de76e268b103d05fa8960e0f77951ff54b012b63429c34f5d6adfd09f5f9ee2 3.5s
=> => extracting sha256:d9a8df5894511ce28a05e2925a75e8a4acbd0634c39ad734fdffa8e23d1b1569 9.8s
=> => extracting sha256:6f51ee005deac0d99898e41b8ce60ebf250ebe1a31a0b03f613aec6bbcb9b3d8 0.1s
=> => extracting sha256:5f32ed3c3f278edda4fc571c880b5277355a29ae8f52b52cdf865f058378a590 5.7s
=> => extracting sha256:0c8cc2f24a4dc6b4e602e086fc9446b0a541e8acd9ad72d2e90df3ba22f158b3 0.3s
=> => extracting sha256:0d27a8e861329007574c6766fba946d48e20d2c8e964e873de352603f22c4ceb 0.1s
=> [internal] load build context
=> => transferring context: 55.57MB 55.9s
=> [2/5] WORKDIR /app 55.7s
=> [3/5] COPY package*.json ./ 0.9s
=> [4/5] RUN npm install 0.4s
=> [5/5] COPY . . 47.1s
=> exporting to image 3.5s
=> => exporting layers 15.8s
=> => exporting manifest sha256:a33ea7d60bea4e4174053cbl87b5ea4a75382bc69e67b2f8ccc2a5a8a2ec8d4 7.1s
=> => exporting config sha256:324d0e23f466b5164adfbf8228dc11c86a7a306ed8b251bd4ecaa99a41ac5a4 0.1s
=> => exporting attestation manifest sha256:c464a0166e10489b2f8509b6c8ad773737d746ba986e1eadd7b9bfcfda415f 0.0s
=> => exporting manifest list sha256:f2026260cfceddfeb4243b98496401af2d07c2951dc5649577387d0a285a7715 0.1s
=> => naming to docker.io/library/myapp:latest 0.0s
=> => unpacking to docker.io/library/myapp:latest 8.2s
```

Exécuter le conteneur Docker : `docker run -p 3000:3000 myapp`

```
C:\express-login-app>docker build -t myapp .
[*] Building 57.3s (11/11) FINISHED
=> [internal] load build definition from Dockerfile
=> => transferring dockerfile: 283B
=> [internal] load metadata for docker.io/library/node:14
=> [internal] load .dockerignore
=> => transferring context: 2B
=> [1/6] FROM docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461aa
=> => resolve docker.io/library/node:14@sha256:a158d3b9b4e3fa813fa6c8c590b8f0a860e015ad4e59bbce5744d2f6fd8461aa
=> [internal] load build context
=> => transferring context: 277.79kB
=> CACHED [2/6] RUN apt-get update && apt-get install -y build-essential python3 && rm -rf /var/lib/apt/list
=> CACHED [3/6] WORKDIR /app
=> [4/6] COPY package*.json ./
=> [5/6] RUN npm install
=> [6/6] COPY . .
=> exporting to image
=> => exporting layers
=> => exporting manifest sha256:c3d3660f24fa14d277ed6206a800a8d9948968b32b601311089996e24678b787
=> => exporting config sha256:10b6477b0b48f539c8429d038b839888750305f281258a3a7f1dba350fe24aaf
=> => exporting attestation manifest sha256:fe5343be3b6e3659e6788f0a4d8a0fb57b8873f0584588230cf5f553b6af745c
=> => exporting manifest list sha256:49f456845d561fa794858e12283af5e3ca106a329ba47cd52c58534069a42283
=> => naming to docker.io/library/myapp:latest
=> => unpacking to docker.io/library/myapp:latest

C:\express-login-app>docker run -p 3000:3000 myapp
http://localhost:3000
MongoDB connected
```

Valider et pousser le Dockerfile :

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS

● PS C:\express-login-app> git add .
● PS C:\express-login-app> git commit -m "Dockerfile added"
[main 5e0cbb1] Dockerfile added
4 files changed, 67 insertions(+), 2189 deletions(-)
create mode 100644 .github/workflows/cd.yml
create mode 100644 .github/workflows/ci.yml
create mode 100644 Dockerfile
delete mode 100644 package-lock.json
● PS C:\express-login-app> git push origin main
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 937 bytes | 234.00 KiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (2/2), completed with 1 local object.
To https://github.com/khawlgr/devops-project.git
7c988b2..5e0cbb1 main -> main
○ PS C:\express-login-app> 
```

📁 .github/workflows	Dockerfile added	1 minute ago
📁 models	Initial commit	19 hours ago
📁 public	first commit	1 hour ago
📁 views	first commit	1 hour ago
📄 .gitignore	first commit	1 hour ago
📄 Dockerfile	Dockerfile added	1 minute ago
📄 README.md	Initial commit	19 hours ago
📄 package.json	first commit	1 hour ago
📄 server.js	first commit	1 hour ago

4. Configurer l'Intégration Continue (CI)

Installer **Mocha** comme dépendance de développement pour les tests dans votre projet.

```
PS C:\express-login-app> npm install --save-dev mocha

up to date, audited 293 packages in 1s

51 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS C:\express-login-app>
```

Enregistrer le workflow CI/CD avec un message de commit, puis le publier sur la branche principale du dépôt distant.

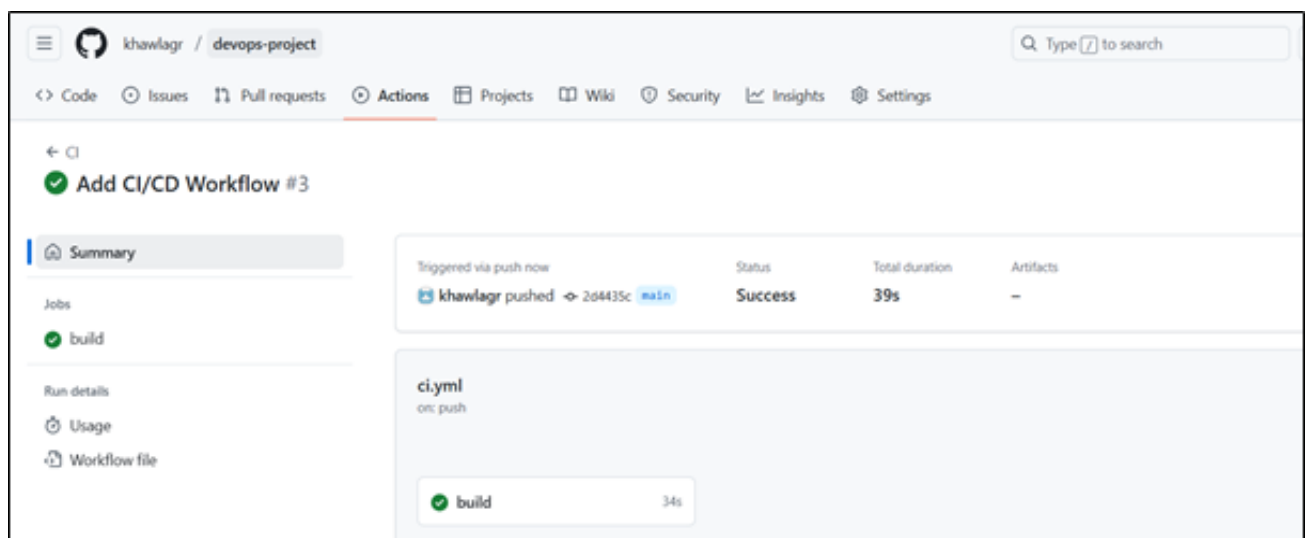
```
git commit -m "Ajout du workflow CI/CD"
git push origin main
```

```
PS C:\express-login-app> git commit -m "Add CI/CD Workflow"
[main 9c2909a] Add CI/CD Workflow
10 files changed, 1547 insertions(+)
create mode 100644 main.tf
create mode 100644 terraform/.terraform.lock.hcl
create mode 100644 terraform/.terraform/providers/registry.terraform.io/kreuzwerker/docker/2.25.0/windows_amd64/CHANGELOG.md
create mode 100644 terraform/.terraform/providers/registry.terraform.io/kreuzwerker/docker/2.25.0/windows_amd64/LICENSE
create mode 100644 terraform/.terraform/providers/registry.terraform.io/kreuzwerker/docker/2.25.0/windows_amd64/README.md
create mode 100644 terraform/.terraform/providers/registry.terraform.io/kreuzwerker/docker/2.25.0/windows_amd64/terraform-provider-docker.exe
create mode 100644 terraform/main.tf
create mode 100644 terraform/terraform.tfstate
create mode 100644 terraform/terraform.tfstate.backup
PS C:\express-login-app> git push origin main
>>
Enumerating objects: 22, done.
Counting objects: 100% (22/22), done.
Delta compression using up to 8 threads
Compressing objects: 100% (14/14), done.
Writing objects: 100% (21/21), 32.99 MiB | 441.00 KiB/s, done.
Total 21 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 1 local object.
remote: warning: See https://gh.io/lfs for more information.
remote: warning: File terraform/bin/terraform.exe is 88.41 MB; this is larger than GitHub's recommended maximum file size
remote: warning: GH001: Large files detected. You may want to try Git Large File Storage - https://git-lfs.github.com.
To https://github.com/khawlagr/devops-project.git
 2d4435c..9c2909a main -> main
PS C:\express-login-app> git push origin main
>>
Everything up-to-date
```

Créer un fichier de configuration **ci.yml**

```
ci.yml
.github > workflows > ci.yml
1
2
3 on:
4   push:
5     branches:
6       - main
7
8 jobs:
9   build:
10    runs-on: ubuntu-latest
11
12    steps:
13      - name: Checkout code
14        uses: actions/checkout@v2
15
16      - name: Set up Docker Buildx
17        uses: docker/setup-buildx-action@v1
18
19      - name: Build Docker image
20        run: docker build -t myapp .
21
22      - name: Run tests
23        run: docker run myapp npm test
24
```

CI ajouté avec succès.

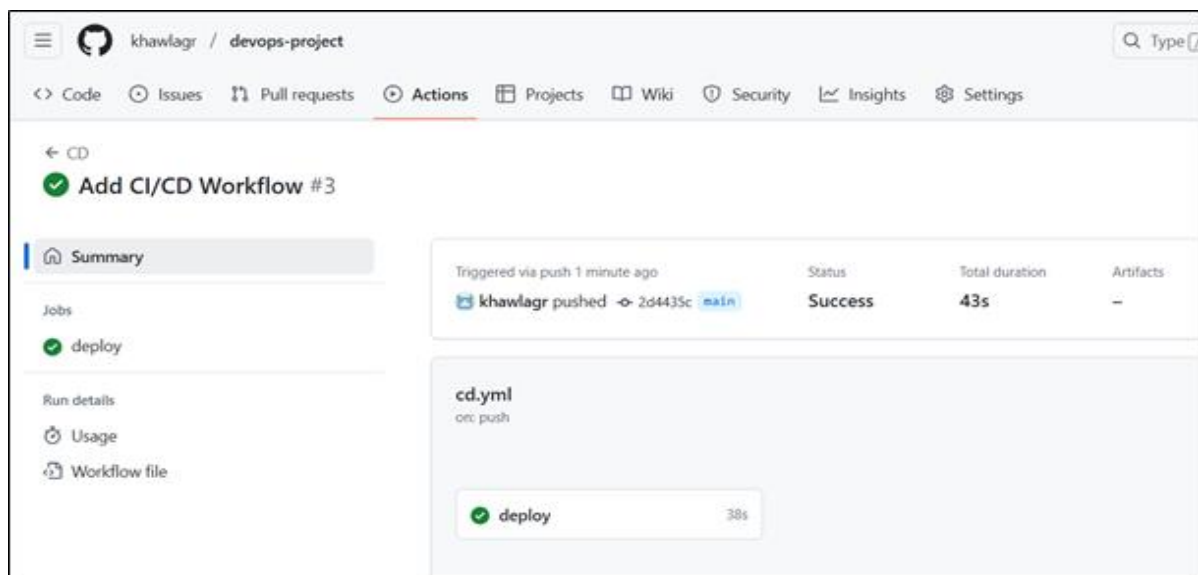


5. Configurer le Déploiement Continu (CD)

Créer un fichier de configuration **cd.yml**

```
cd.yml
.github > workflows > cd.yml
1  name: CD
2
3  on:
4    push:
5      branches:
6        - main
7
8  jobs:
9    deploy:
10     runs-on: ubuntu-latest
11
12     steps:
13     - name: Checkout code
14       uses: actions/checkout@v2
15
16     - name: Set up Docker Buildx
17       uses: docker/setup-buildx-action@v1
18
19     - name: Build Docker image
20       run: docker build -t myapp .
21
22     - name: Deploy Docker container
23       run: |
24         docker stop myapp || true
25         docker rm myapp || true
26         docker run -d -p 3000:3000 --name myapp myapp
27
```

CD ajouté avec succès.



Vérifier l'installation de Terraform avec la commande **terraform -version**

```
Invite de commandes
Microsoft Windows [version 10.0.22000.3260]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\Lenovo>terraform -version
Terraform v1.11.4
on windows_amd64
```

main.tf file:

```
main.tf
1 terraform {
2   required_providers {
3     docker = {
4       source = "kreuzwerker/docker"
5       version = "~> 2.0"
6     }
7   }
8 }
9
10 provider "docker" {}
11
12
13
14 resource "docker_image" "myapp" {
15   name = "myapp"
16   build {
17     context = "${path.module}/.."
18   }
19 }
20
21 resource "docker_container" "myapp" {
22   name = "myapp"
23   image = docker_image.myapp.latest
24
25   ports {
26     internal = 3000
27     external = 3001
28   }
29 }
30
```

Ce fichier déclare un provider Docker, construit une image Docker, puis déploie un conteneur exposant le port 3000 en interne et le port 3001 en externe.

Initialiser Terraform avec la commande : **terraform init**

```
C:\Windows\System32\cmd.exe
Microsoft Windows [version 10.0.22000.3260]
(c) Microsoft Corporation. Tous droits réservés.

C:\express-login-app\terraform>terraform init
Initializing the backend...
Initializing provider plugins...
- Finding kreuzwerker/docker versions matching "~> 2.0"...
- Installing kreuzwerker/docker v2.25.0...
- Installed kreuzwerker/docker v2.25.0 (self-signed, key ID BD080C4571C6104C)
Partner and community providers are signed by their developers.
If you'd like to know more about provider signing, you can read about it here:
https://developer.hashicorp.com/terraform/cli/plugins/signing
Terraform has created a lock file .terraform.lock.hcl to record the provider
selections it made above. Include this file in your version control repository
so that Terraform can guarantee to make the same selections by default when
you run "terraform init" in the future.

Terraform has been successfully initialized!

You may now begin working with Terraform. Try running "terraform plan" to see
any changes that are required for your infrastructure. All Terraform commands
should now work.

If you ever set or change modules or backend configuration for Terraform,
rerun this command to reinitialize your working directory. If you forget, other
commands will detect it and remind you to do so if necessary.

C:\express-login-app\terraform>
```

Appliquer la configuration avec la commande : **terraform apply**

```
C:\express-login-app\terraform>terraform apply
docker_image.myapp: Refreshing state... [id=sha256:e3a1a5a5fe732fbc007660dbfe2407da0b3adb578181edf646de4423c0685d14myapp]
docker_container.myapp: Refreshing state... [id=33a0c769b13d1b7056cd763a3876cb6bc214d6c99ff6cefccc4fba85b14cd4bc]

Terraform used the selected providers to generate the following execution plan. Resource actions are indicated with the
following symbols:
  + create

Terraform will perform the following actions:

# docker_container.myapp will be created
+ resource "docker_container" "myapp" {
  + attach                = false
  + bridge                = (known after apply)
  + command               = (known after apply)
  + container_logs        = (known after apply)
  + container_read_refresh_timeout_milliseconds = 15000
  + entrypoint            = (known after apply)
  + env                   = (known after apply)
  + exit_code             = (known after apply)
  + gateway               = (known after apply)
  + hostname              = (known after apply)
  + id                    = (known after apply)
  + image                  = "sha256:e3a1a5a5fe732fbc007660dbfe2407da0b3adb578181edf646de4423c0685d14"
  + init                  = (known after apply)
  + ip_address            = (known after apply)
  + ip_prefix_length      = (known after apply)
  + ipc_mode              = (known after apply)
  + log_driver            = (known after apply)
  + logs                  = false
  + must_run              = true
  + name                  = "myapp"
  + network_data          = (known after apply)
  + read_only             = false
  + remove_volumes       = true
  + restart               = "no"
  + rm                    = false
```

```

+ healthcheck (known after apply)

+ labels (known after apply)

+ ports {
  + external = 3000
  + internal = 3000
  + ip       = "0.0.0.0"
  + protocol = "tcp"
}
}
}

Plan: 1 to add, 0 to change, 0 to destroy.

Warning: Deprecated attribute
  on main.tf line 23, in resource "docker_container" "myapp":
   23:   image = docker_image.myapp.latest

The attribute "latest" is deprecated. Refer to the provider documentation for details.
(and one more similar warning elsewhere)

Do you want to perform these actions?
  Terraform will perform the actions described above.
  Only 'yes' will be accepted to approve.

Enter a value: yes

docker_container.myapp: Creating...
docker_container.myapp: Still creating... [10s elapsed]
docker_container.myapp: Creation complete after 10s [id=a137806d3b5c36f2e416c9e0076dfa631b6f6a84e4e12ba931d4f763e8a32670]

Warning: Deprecated attribute
  on main.tf line 23, in resource "docker_container" "myapp":
   23:   image = docker_image.myapp.latest

The attribute "latest" is deprecated. Refer to the provider documentation for details.
(and one more similar warning elsewhere)

Apply complete! Resources: 1 added, 0 changed, 0 destroyed.

```

Nous vous remercions pour votre attention ☺