

CANARIO Théo

Software engineering for the cloud

## 1. Creation de la mini app (powered by ChatGPT)

```
project — -zsh — 158x52
theoc@MacBook-Pro-de-Theo-2 cloud % mkdir project
theoc@MacBook-Pro-de-Theo-2 cloud % cd project
theoc@MacBook-Pro-de-Theo-2 project % npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See 'npm help init' for definitive documentation on these fields
and exactly what they do.

Use 'npm install <pkg>' afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (project) project-cloud
version: (1.0.0)
description: a simple project for cloud enigneering :)
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
About to write to /Users/theoc/Documents/efrei/S9/cloud/project/package.json:

{
  "name": "project-cloud",
  "version": "1.0.0",
  "description": "a simple project for cloud enigneering :)",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}

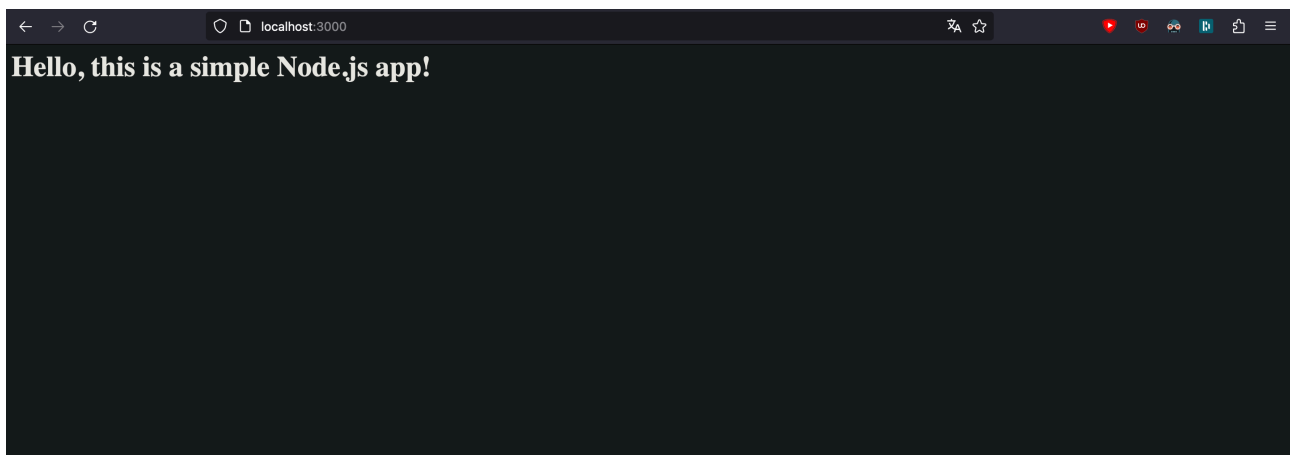
Is this OK? (yes)
npm notice
npm notice New major version of npm available! 8.5.0 -> 10.2.4
npm notice Changelog: https://github.com/npm/cli/releases/tag/v10.2.4
npm notice Run npm install -g npm@10.2.4 to update!
npm notice
theoc@MacBook-Pro-de-Theo-2 project % npm install express

added 62 packages, and audited 63 packages in 3s

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

found 0 vulnerabilities
theoc@MacBook-Pro-de-Theo-2 project %
```

```
JS app.js ×
Users > theoc > Documents > efrei > S9 > cloud > project > JS app.js > ...
1 // app.js
2 const express = require('express');
3 const app = express();
4 const port = 3000;
5
6 app.get('/', (req, res) => {
7   res.send('<h1>Hello, this is a simple Node.js app!</h1>');
8 });
9
10 app.listen(port, () => {
11   console.log(`App listening at http://localhost:\${port}`);
12 });
13
```



## 2. Creation of the docker file

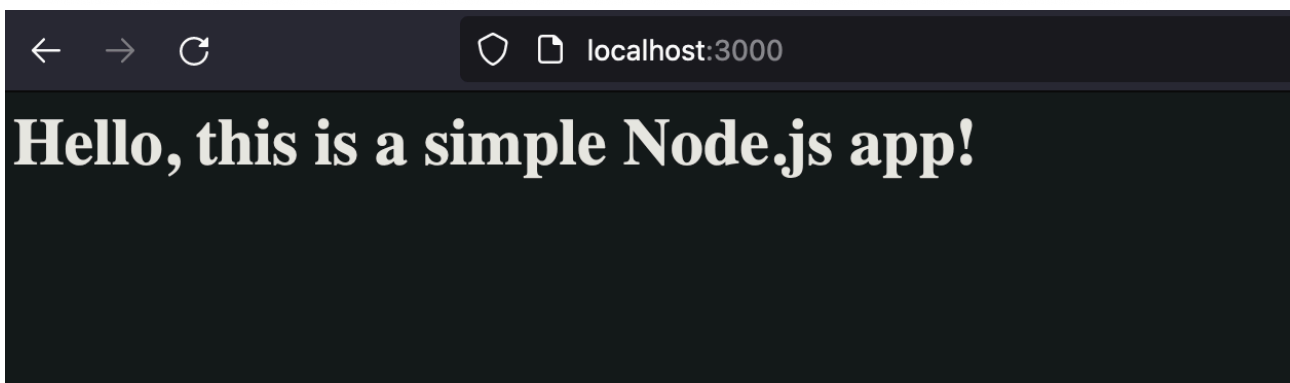
```
JS app.js Dockerfile ●
Users > theoc > Documents > efrei > S9 > cloud > project > Dockerfile > ...
1  # Use the official Node.js image as the base image
2  FROM node:14
3
4  # Set the working directory in the container
5  WORKDIR /usr/src/app
6
7  # Copy package.json and package-lock.json to the container
8  COPY package*.json ./
9
10 # Install app dependencies
11 RUN npm install
12
13 # Copy the rest of the application code to the container
14 COPY . .
15
16 # Expose the port on which the app will run
17 EXPOSE 3000
18
19 # Command to run the application
20 CMD ["node", "app.js"]
21
```

3. Build docker app : `docker build -t simple-node-app:1.0 .`

4. Run docker app : `docker run -p 3000:3000 simple-node-app:1.0`

```
project — com.docker.cli - docker run -p 3000:3000 simple-node-app:1.0 — 132x28
=> sha256:1de76e268b103d05fa8960e0f77951ff54b912b63429c34f5d6adfd09f5f9ee2 51.88MB / 51.88MB 31.4s
=> sha256:d9a8df5894511ce28a05e2925a75e8a4acbd0634c39ad734fd8ba8e23d1b1569 191.85MB / 191.85MB 52.1s
=> sha256:6f51ee005deac0d99898e41b8ce60ebf250ebe1a31a0b03f613aec6bbcb9b83d8 4.19kB / 4.19kB 13.3s
=> extracting sha256:2ff1d7c41c74a25258bfa6f0b8adb0a727f84518f55f65ca845ebc747976c408 3.1s
=> sha256:5f32ed3c3f278edda4fc571c880b5277355a29ae8f52b52cdf865f058378a590 35.24MB / 35.24MB 27.0s
=> extracting sha256:b253aaafeaa7e0671bb60008df01de101a38a045ff7bc656e3b0fbfc7c05cca5 0.3s
=> extracting sha256:3d2201bd995ccc12851a50820de03d34a17011dcb9ac9fd3a50c952cbb131 0.3s
=> sha256:0c8cc2f24a4dcb64e602e086fc9446b0a541e8acd9ad72d2e90df3ba22f158b3 2.29MB / 2.29MB 29.9s
=> sha256:0d27a8e861329007574c6766fba946d48e20d2c8e964e873de352603f22c4ceb 450B / 450B 30.3s
=> extracting sha256:1de76e268b103d05fa8960e0f77951ff54b912b63429c34f5d6adfd09f5f9ee2 3.6s
=> extracting sha256:d9a8df5894511ce28a05e2925a75e8a4acbd0634c39ad734fd8ba8e23d1b1569 9.4s
=> sha256:6f51ee005deac0d99898e41b8ce60ebf250ebe1a31a0b03f613aec6bbcb9b83d8 0.1s
=> extracting sha256:5f32ed3c3f278edda4fc571c880b5277355a29ae8f52b52cdf865f058378a590 3.1s
=> extracting sha256:0c8cc2f24a4dcb64e602e086fc9446b0a541e8acd9ad72d2e90df3ba22f158b3 0.2s
=> extracting sha256:0d27a8e861329007574c6766fba946d48e20d2c8e964e873de352603f22c4ceb 0.0s
=> [2/5] WORKDIR /usr/src/app 0.9s
=> [3/5] COPY package*.json ./ 0.0s
=> [4/5] RUN npm install 3.4s
=> [5/5] COPY . . 0.1s
=> exporting to image 0.1s
=> exporting layers 0.1s
=> writing image sha256:755f01e1cd5d672c5eb3919bb24c35b740d6941a828c44fef9872fbdd7e85c69 0.0s
=> naming to docker.io/library/simple-node-app:1.0 0.0s

Use 'docker scan' to run Snyk tests against images to find vulnerabilities and learn how to fix them
[theoc@MacBook-Pro-de-Theo-2 project % docker run -p 3000:3000 simple-node-app:1.0
App listening at http://localhost:3000
]
```



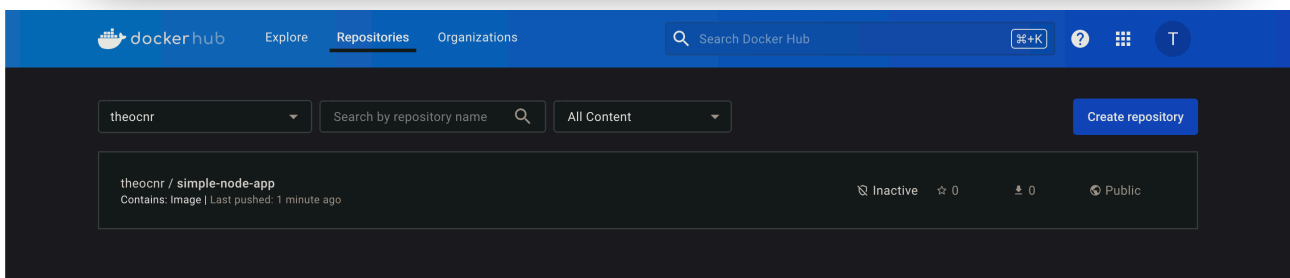
## 5. Publish the docker image to the docker hub

```
theoc — -zsh — 129x31
Login with your Docker ID to push and pull images from Docker Hub. If you don't have a Docker ID, head over to https://hub.docker.com to create one.
Username: TheoCnr
Password:
Error response from daemon: Get "https://registry-1.docker.io/v2/": unauthorized: incorrect username or password
theoc@MBP-de-Theo-2 ~ % docker login -u theocnr
Password:
Login Succeeded

Logging in with your password grants your terminal complete access to your account.
[For better security, log in with a limited-privilege personal access token. Learn more at https://theotheocetheocethtttttttttt]
theoc@MBP-de-Theo-2 ~ % docker tag simple-node-app:1.0 theocnr/simple-node-app:1.0

[theoc@MBP-de-Theo-2 ~ % docker push theocnr/simple-node-app:1.0

The push refers to repository [docker.io/theocnr/simple-node-app]
4b4b3b468129: Pushed
17f5a3ae0de2: Pushed
fdd0920bfa49: Pushed
361f43bf2064: Pushed
0d5f5a015e5d: Mounted from library/node
3c777d951de2: Mounted from library/node
f8a91dd5fc84: Mounted from library/node
cb81227abde5: Mounted from library/node
e01a454893a9: Mounted from library/node
c45660adde37: Mounted from library/node
fe0fb3ab4a0f: Mounted from library/node
f1186e5061f2: Mounted from library/node
b2dba7477754: Mounted from library/node
1.0: digest: sha256:bd15c53ee3e7c62c5ab7fdaf3f5963ed30f912979d27935f75157b770df633c2 size: 3051
theoc@MBP-de-Theo-2 ~ % █
```



## 6. Create a kubernetes deployment

```
theoc@MBP-de-Theo-2 ~ % minikube start
minikube v1.31.2 sur Darwin 12.1
minikube 1.32.0 est disponible ! Téléchargez-le ici : https://github.com/kubernetes/minikube/releases/tag/v1.32.0
Pour désactiver cette notification, exécutez : 'minikube config set WantUpdateNotification false'

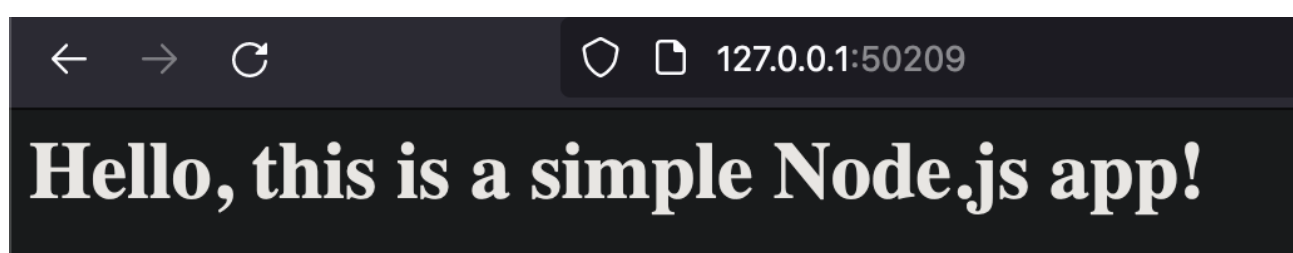
! Utilisation du pilote docker basé sur le profil existant
! Démarrage du noeud de plan de contrôle minikube dans le cluster minikube
! Extraction de l'image de base...
! docker "minikube" container est manquant, il va être recréé.
! Création de docker container (CPUs=2, Memory=2200Mo) ...
! Préparation de Kubernetes v1.27.4 sur Docker 24.0.4...
! Configuration de bridge CNI (Container Networking Interface)...
! Vérification des composants Kubernetes...
! Utilisation de l'image gcr.io/k8s-minikube/storage-provisioner:v5
! Modules activés: storage-provisioner, default-storageclass

! /usr/local/bin/kubectl est la version 1.25.0, qui peut comporter des incompatibilités avec Kubernetes 1.27.4.
! Vous voulez kubectl v1.27.4 ? Essayez 'minikube kubectl -- get pods -A'
Terminé ! kubectl est maintenant configuré pour utiliser "minikube" cluster et espace de noms "default" par défaut.
theoc@MBP-de-Theo-2 ~ %
```

```
! deployment.yaml
1  apiVersion: apps/v1
2  kind: Deployment
3  metadata:
4    name: simple-node-app
5  spec:
6    replicas: 1
7    selector:
8      matchLabels:
9        app: simple-node-app
10   template:
11     metadata:
12       labels:
13         app: simple-node-app
14     spec:
15       containers:
16       - name: simple-node-app
17         image: theocnr/simple-node-app:1.0
18         ports:
19         - containerPort: 3000
20
```

```
project — ssh < minikube service simple-node-app --url — 129x31

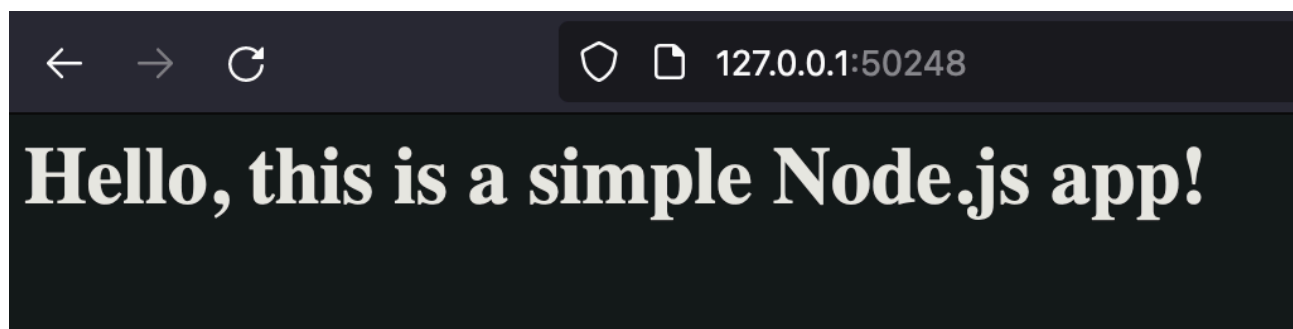
theoc@MBP-de-Theo-2 project % kubectl apply -f deployment.yaml
deployment.apps/simple-node-app created
theoc@MBP-de-Theo-2 project % kubectl get deployments
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
myservice     1/1     1            1           70d
simple-node-app 0/1     1            0           11s
theoc@MBP-de-Theo-2 project % kubectl expose deployment simple-node-app --type=NodePort --port=3000
service/simple-node-app exposed
theoc@MBP-de-Theo-2 project % minikube service simple-node-app --url
http://127.0.0.1:50209
! Comme vous utilisez un pilote Docker sur darwin, le terminal doit être ouvert pour l'exécuter.
```



## 7. Create kubernetes service

```
! service.yaml
1  apiVersion: v1
2  kind: Service
3  metadata:
4    name: simple-node-app-service
5  spec:
6    selector:
7      app: simple-node-app
8    ports:
9      - protocol: TCP
10        port: 80
11        targetPort: 3000
12    type: LoadBalancer
```

```
[theo@MBP-de-Theo-2 project % kubectl apply -f service.yaml
service/simple-node-app-service created
[theo@MBP-de-Theo-2 project % kubectl get services
NAME                                TYPE                CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
kubernetes                         ClusterIP           10.96.0.1       <none>           443/TCP          70d
simple-node-app                    NodePort            10.110.153.17   <none>           3000:31637/TCP   7m55s
simple-node-app-service            LoadBalancer       10.96.121.199   <pending>        80:31320/TCP     9s
[theo@MBP-de-Theo-2 project % kubectl get services
NAME                                TYPE                CLUSTER-IP      EXTERNAL-IP      PORT(S)          AGE
kubernetes                         ClusterIP           10.96.0.1       <none>           443/TCP          70d
simple-node-app                    NodePort            10.110.153.17   <none>           3000:31637/TCP   8m17s
simple-node-app-service            LoadBalancer       10.96.121.199   <pending>        80:31320/TCP     31s
[theo@MBP-de-Theo-2 project % minikube service simple-node-app-service --url
http://127.0.0.1:50248
! Comme vous utilisez un pilote Docker sur darwin, le terminal doit être ouvert pour l'exécuter.
```

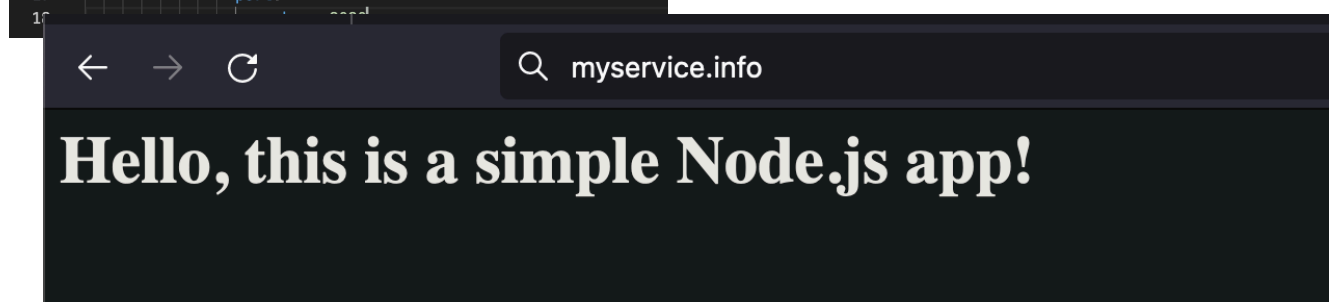


## 8. Add service

We use the ingress.yaml file provided in the GitHub.

```
[theo@macbook-pro-de-theo project % kubectl apply -f ingress.yaml
ingress.networking.k8s.io/example-ingress created
[theo@macbook-pro-de-theo project % kubectl get ingress example-ingress
NAME          CLASS    HOSTS          ADDRESS    PORTS    AGE
example-ingress <none>    myservice.info          80       86s
[theo@macbook-pro-de-theo project %
```

```
Users > theo > Documents > enri > S9 > cloud > project > ! ingress.yaml
1  apiVersion: networking.k8s.io/v1
2  kind: Ingress
3  metadata:
4    name: example-ingress
5    annotations:
6      # nginx.ingress.kubernetes.io/rewrite-target: /$1
7  spec:
8    rules:
9      - host: myservice.info
10        http:
11          paths:
12            - path: /
13              pathType: Prefix
14              backend:
15                service:
16                  name: myservice
17                  port:
18                    number: 80
```



## 9. Add a second service

I used the file front-back-app.yaml, I didn't really understand this part so it is not completed but this is my work so far.

Then I used :

```
kubectl apply -f front-back-app.yaml
```

But I can't access it on browser, I'm still wondering why

Cours	Atelier	Quête	Quiz	Jeu	En cours	Terminée
Activité	Type	Date de début	Date de fin	Score	Réussie	
<a href="#">Kubernetes Engine : Qwik Start</a>	Atelier	14 nov. 2023	14 nov. 2023	Assessment: 100%	✓	
<a href="#">Introduction to Docker</a>	Atelier	8 nov. 2023	8 nov. 2023	Assessment: 100%	✓	
<a href="#">Configuring Networks via gcloud</a>	Atelier	8 nov. 2023	8 nov. 2023	Assessment: 100%	✓	
<a href="#">Créer une machine virtuelle</a>	Atelier	7 nov. 2023	7 nov. 2023	Assessment: 100%	✓	
<a href="#">Présentation des ateliers pratiques Google Cloud</a>	Atelier	7 nov. 2023	7 nov. 2023	Assessment: 100%	✓	