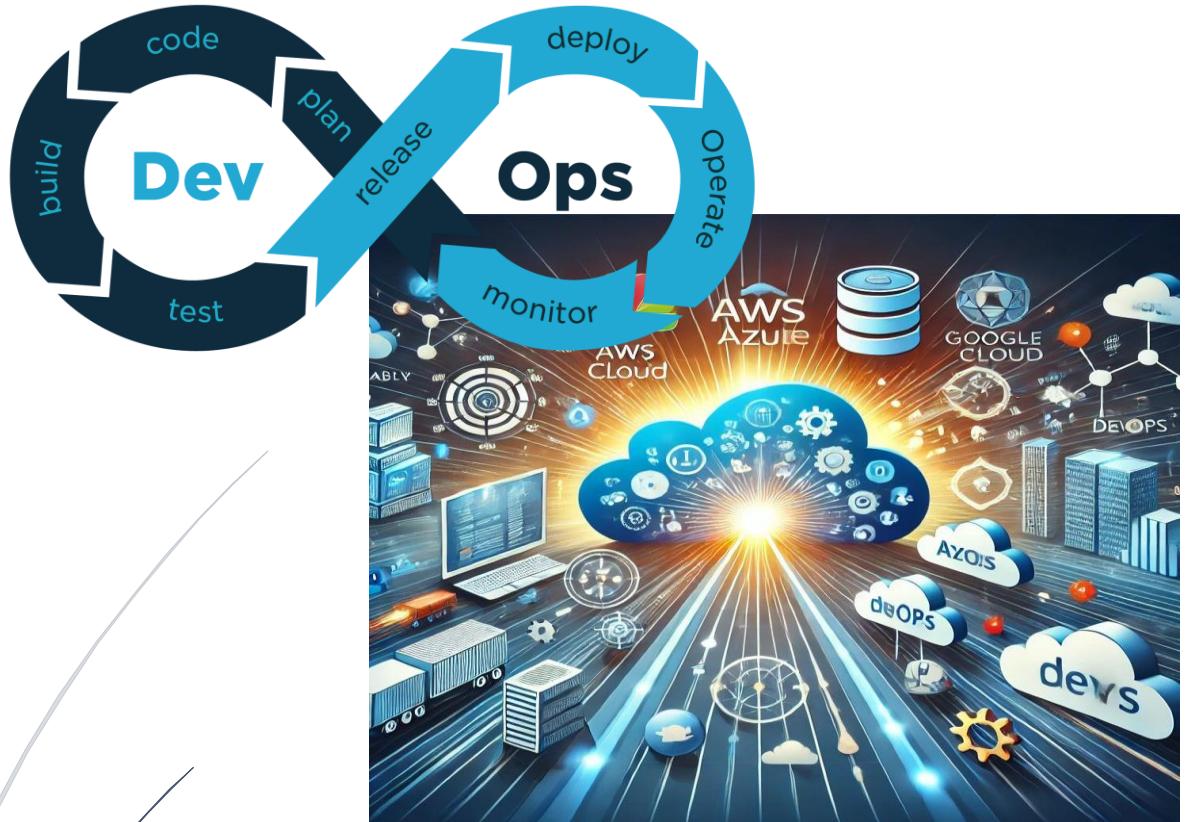




EVALUATION EN COURS DE FORMATION

PARCOURS ADMINISTRATEUR SYSTEME DEVOPS



Préparer un environnement de test

Evaluation : ECF5

Version : du 10/11/2025

Auteur : FT

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

Sommaire

Table des matières

1. PRESENTATION	2
1.1. DESCRIPTION DE LA COMPETENCE – PROCESSUS DE MISE EN ŒUVRE	2
1.2. CONTEXTE(S) PROFESSIONNEL(S) DE MISE EN ŒUVRE.....	2
1.3. CRITERES DE PERFORMANCE.....	2
1.4. SAVOIR-FAIRE TECHNIQUES, ORGANISATIONNELS, RELATIONNELS	2
1.5. CAHIER DES CHARGES	3
2. ECF1 - QUESTIONS.....	4
2.1. POURQUOI VAUT-IL MIEUX TRAVAILLER SUR UN AUTRE ENVIRONNEMENT QUE CELUI DE PRODUCTION ?	4
2.2. QU'EST CE QU'ANSIBLE ?	4
2.3. QU'EST-CE QU'UN REVERSE PROXY ?	6
2.4. POURQUOI UTILISER DIFFERENTES BRANCHES SUR LE MEME REPO GIT ?	7
3. CREATION DE L'INFRASTRUCTURE GIT ET TERRAFORM.....	8
3.1. PREREQUIS.....	8
3.2. CREATION DE LA BRANCHE "TEST" SUR GIT ET GITHUB	8
3.3. CREATION DE L'ENVIRONNEMENT TERRAFORM CLOUD	9
4. CREATION DE L'APPLICATION NODE.JS	12
4.1. INITIALISATION	12
4.2. CREATION DU FICHIER "INDEX.JS"	13
4.3. TEST LOCAL	13
5. CREATION DES SCRIPTS TERRAFORM	14
5.1. SCRIPT MAINT.TF (EN ANNEXE)	14
5.2. DECLARATION DES VARIABLES	16
5.3. SCRIPT BASH (INSTALLATION ANSIBLE)	17
6. CREATION DU PLAYBOOK ANSIBLE (EN ANNEXE).....	18
7. COMMIT DES BRANCHES TEST ET MAIN.....	20
8. LANCEMENT DES RUNS TERRAFORM	22
8.1. RUN TEST	22
8.2. RUN PROD.....	24
9. ANNEXE	26
9.1. SCRIPT TERRAFORM.....	26
9.2. SCRIPT ANSIBLE.....	28

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

1. PRESENTATION

1.1. DESCRIPTION DE LA COMPETENCE – PROCESSUS DE MISE EN ŒUVRE

- A la demande d'un client ou de son responsable, mettre en place un environnement de test.
- En tenant compte d'un cahier des charges.
- Le mettre en service et documenter son travail.

1.2. CONTEXTE(S) PROFESSIONNEL(S) DE MISE EN ŒUVRE

Préparer un environnement de développement et le mettre à disposition des développeurs

1.3. CRITERES DE PERFORMANCE

- L'environnement de tests est conforme au cahier des charges
- Une version de test de l'application est produite
- Les tests sont effectués
- Les dysfonctionnements sont remontés aux développeurs

1.4. SAVOIR-FAIRE TECHNIQUES, ORGANISATIONNELS, RELATIONNELS

- Utiliser un outil de gestion des versions de code (de type Github)
- Créer une infrastructure de test adaptée au projet
- Récupérer les codes de l'application à mettre en production
- Déployer l'application dans l'environnement de test
- Effectuer les premiers tests
- Anticiper les changements nécessaires sur l'infrastructure pour la nouvelle version
- Faire un retour aux développeurs sur les dysfonctionnements constatés sur la version en test
- Diagnostiquer un dysfonctionnement lié à l'infrastructure et le corriger
- Echanger avec les développeurs
- Consulter de la documentation technique rédigée en anglais
- Effectuer une veille technologique
- Connaissance des outils de gestion de version de code
- Connaissance de la démarche DevOps
- Comprendre l'organisation des équipes de développement
- Comprendre la méthode Agile
- Connaissance de la démarche CI/CD

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

1.5. CAHIER DES CHARGES

- **IMPORTANT** : Toujours Destroy après avoir Apply pour ne pas payer quoi que ce soit.
- Le cahier des charges est réalisé avec une terminologie AWS.
- Création branche "**test**" sur le VCS et une "**Organisation Test**" sur **Terraform Cloud**
- Dans l'organisation test créer un **Workspace** connecté à la branche "**test**" du repo
- Créer une application **node.js**
 - Faire d'abord tourner l'application sur la machine pour être sûr qu'elle marche comme souhaité
 - <https://nodejs.org/en/docs/guides/getting-started-guide/>
 - <https://nodejs.org/en/download/>
 - Faire en sorte que la réponse à la requête "/" contient la variable d'environnement **ENVIRONMENT_NAME**.
Voir le paquet **npm** nommée "**dotenv**"
- Provisionner un **playbook Ansible** dépendant de l'instance qui devra : (il faudra sûrement installer Ansible et lancer TFC en mode local)
 - Update et upgrade les paquets à l'initialisation
 - Installer **nginx** et **nodejs**
 - Créer un fichier de configuration nginx mettant en place un **reverse proxy** redirigeant les requêtes reçus au **port 80** vers **localhost :3000**
 - Lancer nginx
 - Ajouter une variable d'environnement nommée **ENVIRONMENT_NAME** qui dans la branche test aura la **valeur test** et sur master (main) la **valeur prod**
 - Lancer l'application avec node.js sur le port 300
 - Lancez les workflows prod et test, vérifier l'accès aux deux sites web, l'un renvoyant un message contenant le mot test et l'autre prod
- **Livrables** : Deux serveurs chacun dans un environnement différent consultable via internet et facilement différenciable

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

2. ECF5 - QUESTIONS

2.1. POURQUOI VAUT-IL MIEUX TRAVAILLER SUR UN AUTRE ENVIRONNEMENT QUE CELUI DE PRODUCTION ?

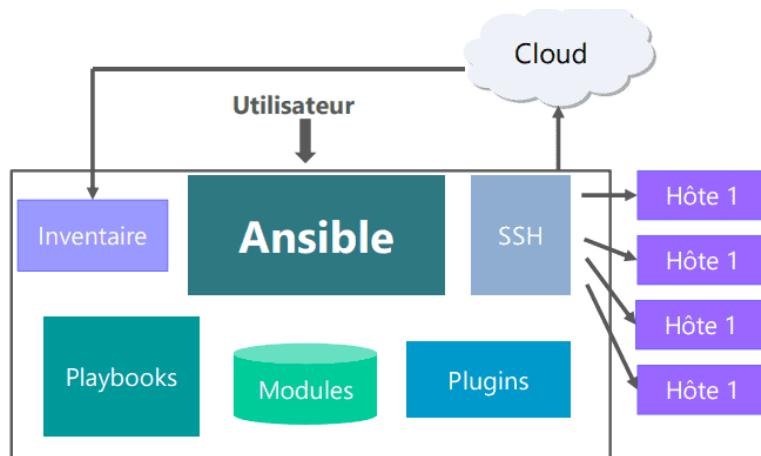
- Comme son nom l'indique, un environnement de développement est une plateforme permettant de tester et de déboguer des éléments.
- En revanche, en production, le produit est rendu public et il est impossible d'afficher des erreurs et des bugs à l'utilisateur final.
- Les raisons sont les suivantes :
 - **Éviter les interruptions de service:**
Toute erreur, tout bug introduit directement peut entraîner des pannes, des dysfonctionnements et une mauvaise expérience utilisateur.
 - **Minimiser les risques de perte de données :**
Sur un environnement de production, une erreur de manipulation pourrait entraîner une perte de données irréversible, avec des conséquences légales et financières importantes.
 - **Faciliter la collaboration et l'expérimentation :**
Différents environnements permettent à plusieurs équipes de travailler en parallèle sans se gêner.
 - **Assurer la stabilité et la fiabilité :**
On s'assure que seules les versions stables et éprouvées sont déployées en production, garantissant ainsi la fiabilité du service.
 - **Simplifier le rollback en cas de problème :**
Il est beaucoup plus simple et rapide de revenir à la version précédente stable si les tests ont été effectués dans un environnement dédié.

2.2. QU'EST CE QU'ANSIBLE ?

- **Ansible est un outil d'automatisation informatique** open source qui simplifie et automatise divers processus informatiques manuels, notamment le provisionnement, la gestion de la configuration, le déploiement d'applications et l'orchestration.
- **Ansible utilise un langage impératif appelé YAML** pour définir les tâches d'automatisation, ce qui le rend facile à lire et à comprendre.

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

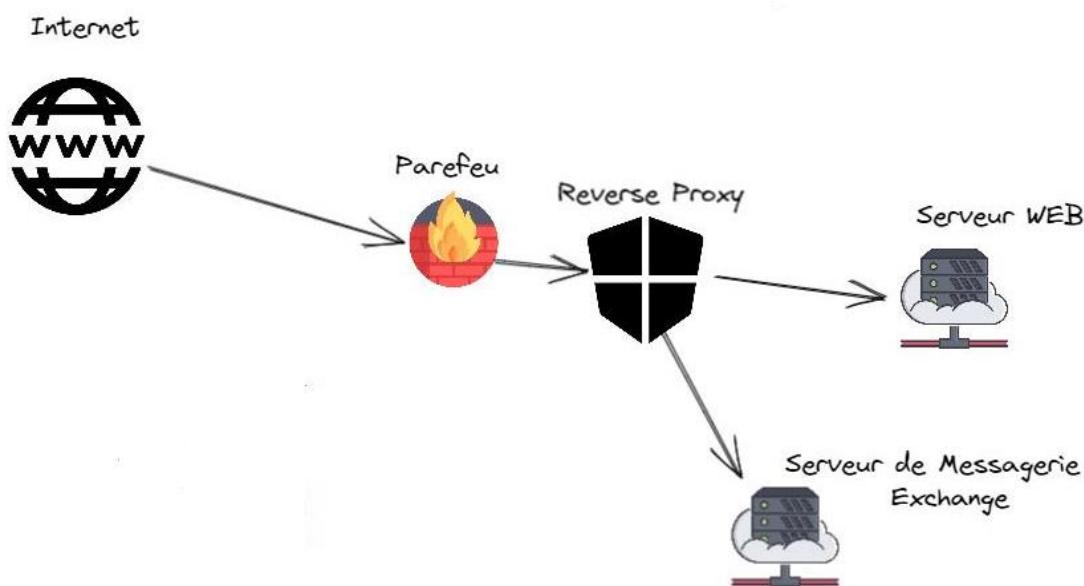
- **Ansible utilise un playbook** qui est un plan de tâches d'automatisation exécuté avec un effort manuel limité sur un ensemble de solutions informatiques.
 - Les playbooks indiquent à Ansible ce qu'il doit faire et comment le faire.
 - Écrits en YAML, ils permettent d'automatiser un large éventail de tâches, de la simple gestion de configuration aux déploiements d'applications complexes.
 - Exécutables sur un seul nœud ou sur plusieurs nœuds simultanément, ils simplifient la gestion de l'infrastructure à grande échelle.
- **Gestion de configuration simplifiée**
 - Simple et **Agentless**
 - **Protocole de connexion** : Ansible utilise principalement SSH pour se connecter aux machines distantes et exécuter des tâches.
- **Termes Ansible**
 - **Inventaire :**
L'inventaire dans Ansible est un fichier (généralement au format YAML) qui répertorie tous les nœuds ou hôtes.
 - **Playbooks :**
Les Playbooks sont des fichiers YAML qui définissent les tâches à exécuter par Ansible sur les différents nœuds de l'inventaire.
 - **Modules :**
Les Modules sont des programmes exécutés par Ansible sur les nœuds distants pour accomplir des tâches spécifiques.
 - **Plugins :**
Les Plugins dans Ansible étendent les fonctionnalités de base du système en ajoutant de nouvelles fonctionnalités, intégrations ou capacités.



Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

2.3. QU'EST-CE QU'UN REVERSE PROXY ?

- Un serveur reverse proxy se place devant les serveurs web et transmet les requêtes des clients à ces serveurs.
- Les ressources demandées sont ensuite renvoyées au client, comme si elles provenaient du serveur proxy lui-même.
- Un reverse proxy offre :
 - **Sécurité renforcée**
Aucune information des serveurs back-end n'est visible en dehors du réseau interne ; les clients malveillants ne peuvent donc pas y accéder directement pour exploiter leurs vulnérabilités.
 - **Évolutivité et flexibilité accrues.**
Comme les clients ne voient que l'adresse IP du proxy inverse, vous êtes libre de modifier la configuration de votre infrastructure back-end.
 - **Accélération Web.**
Les reverse proxys peuvent réduire le temps nécessaire à la génération d'une réponse et à son retour au client afin d'améliorer les performances grâce à des techniques telles que l'accélération SSL, la compression intelligente et la mise en cache.



En résumé, un reverse proxy est un serveur qui se place devant un ou plusieurs serveurs web et agit comme un point de contact unique pour les clients.

Il offre de nombreux avantages en termes de sécurité, de performance et de gestion.

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

2.4. POURQUOI UTILISER DIFFERENTES BRANCHES SUR LE MEME REPO GIT ?

- **Définition**

- Une branche est une ligne de développement indépendante contenant un pointeur vers le commit le plus récent dans le code.
- Un dépôt Git unique peut suivre plusieurs branches, mais le travail est centré sur une seule. L'en-tête pointe vers le dernier commit d'une branche.
- Une branche est automatiquement générée lors de la création d'un dépôt du code.

Cette branche par défaut s'appelle "main".

L'écriture de code et la validation du travail enregistrent automatiquement une validation dans la branche de travail.

- **Intérêt d'une branche**

- Une branche est souvent utilisée pour les nouvelles fonctionnalités et la correction de bugs.
- Elle permet de travailler sans impacter la base de code principale.
 - Une fois le travail terminé, on peut intégrer les modifications à l'application en fusionnant la branche avec la base de code principale.
 - Cela permet à plusieurs personnes de travailler simultanément sur différents aspects du projet.
 - Cela permet également d'expérimenter de nouvelles idées sans compromettre la stabilité du code principal.

- **En résumé,** l'utilisation de branches dans Git permet de :

- Organiser le travail de développement.
- Isoler les changements et les risques.
- Faciliter la collaboration en équipe.
- Gérer différentes versions du projet.

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

3. CREATION DE L'INFRASTRUCTURE GIT ET TERRAFORM

3.1. PREREQUIS

- Clone du projet ECF3 :

Un clone de l'évaluation ECF3 sera faite dans le dossier ec5 pour cet exercice.

Un clone de l'évaluation ECF3 sera faite dans le dossier ec5 pour cet exercice.	Ce PC > Disque local (C:) > devops Nom Modifié le EC3 10/05/2025 11 EC4 10/05/2025 10 EC5 10/05/2025 17 Fichier Accueil Partage Affichage Desktop Téléchargement Documents Images EC5 ECF3 - Sécuriser ECF4 - Mettre l'infra ECF5 - Préparer l'infra Ce PC Nom .git modules .gitignore .vault-token main.tf README.md terraform.tfvars variables.tf
---	---

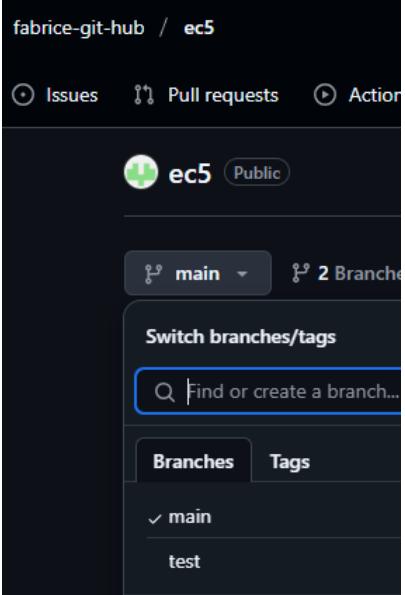
- Créé durant l'évaluation ECF3 :
 - Un compte [Terraform Cloud](#)
 - Un compte [AWS](#)
 - Une clé SSH (.pem) pour accéder aux instances EC2

3.2. CREATION DE LA BRANCHE "TEST" SUR GIT ET GITHUB

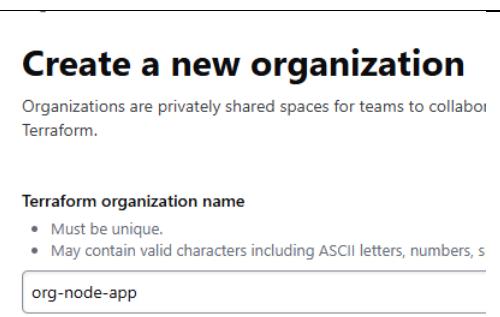
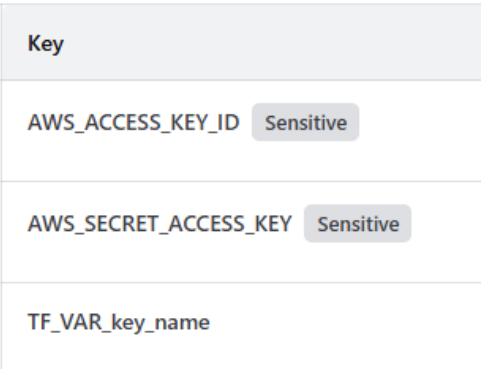
Le dépôt GitHub contiendra le projet : <https://github.com/fabrice-git-hub/ec5.git>

git clone https://github.com/fabrice-git-hub/ec3.git C:\EC5	\$ git clone https://github.com/fabrice-git-hub/ec3.git c:\EC5 Cloning into 'C:\EC5'... remote: Enumerating objects: 9, done. remote: Counting objects: 100% (9/9), done. remote: Compressing objects: 100% (7/7), done. remote: Total 9 (delta 0), reused 9 (delta 0), pack-reused 0 (from 0) Receiving objects: 100% (9/9), done.
git checkout -b test	Création de la branche test

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

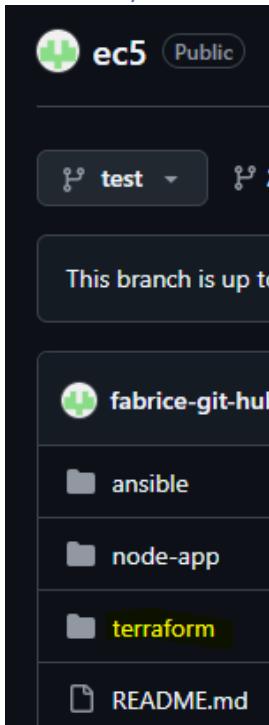
Push de la branche locale "test" vers le repository distant <code>git push origin test</code>	
--	--

3.3. CREATION DE L'ENVIRONNEMENT TERRAFORM CLOUD

• Création de l'Organisation Terraform : org-node-app	
• Création du SET de Variables au niveau de l'Organisation <ul style="list-style-type: none"> ○ Variables pour l'accès à AWS : AWS_ACCESS_KEY_ID AWS_SECRET_ACCESS_KEY ○ Variable pour la clé SSH qui contiendra le nom de la paire de clé AWS : TF_VAR_key_name 	

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

- Création du Workspace pour chaque Branche
- Se connecter au Repository **EC5**
- Répertoire de travail : [/terraform](#)



Create a new Workspace

HCP Terraform organizes your infrastructure resources by workspace. A workspace contains infrastructure resources, variables, state data, and workspaces in HCP Terraform.

Choose your workflow

Version Control Workflow

Trigger runs based on changes to configuration in repositories.

Best for those who need traceability and transparency

Choose a repository

Choose the repository that hosts your Terraform source code requests.

Don't have a repo? Here's an [example repo](#) you can use.

[fabricice-github](#) 1 repository

ec5

Workspace Settings

Terraform Working Directory

/terraform

- Création du Workspace connecté à la branche "test" : **TEST**

Create a new Workspace

HCP Terraform organizes your infrastructure resources by workspace. A workspace contains infrastructure resources, variables, state data, and workspaces in HCP Terraform.

Configure Settings

Workspace Name

TEST

The name of your workspace is unique and used in tools, such as Terraform. Only alphanumeric characters are permitted. Learn more about workspace names.

Description (Optional)

Branche de TEST du repository [Github EC5](#)

VCS branch

test

The branch from which HCP Terraform will control provides as the working directory for the workspace.

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

- Création du Workspace connecté à la branche "main" : PROD

Create a new Workspace

HCP Terraform organizes your infrastructure resources by workspace. A workspace contains infrastructure resources, variables, state data, and more. You can have multiple workspaces in HCP Terraform.

Configure Settings

Workspace Name

PROD

VCS branch

main

The branch from control provides

The name of your workspace is unique and used in tools, runbooks, and reports. Only alphanumeric characters are permitted. Learn more about [repository branches](#).

Description (Optional)

Branche de PROD du repository [Github](#) ECS

- Création de la Variables pour chaque environnement

ENVIRONMENT_NAME (test ou prod)

Workspace variables (1)

Variables defined within a workspace always overwrite variables from variable sets or environment variables.

Key	Value
ENVIRONMENT_NAME	PROD

Workspace variables (1)

Variables defined within a workspace always overwrite variables from variable sets or environment variables.

Key	Value
ENVIRONMENT_NAME	TEST

org-node-app / Workspaces

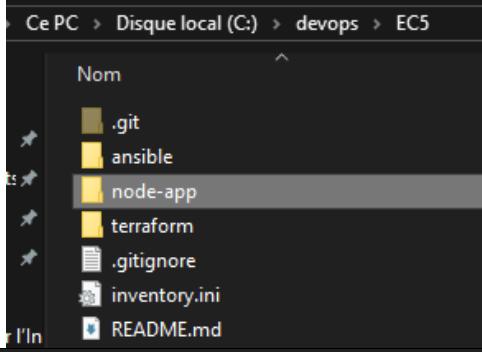
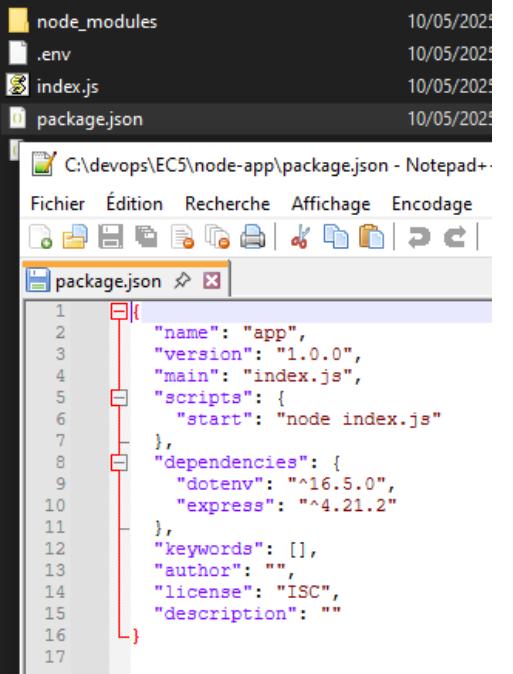
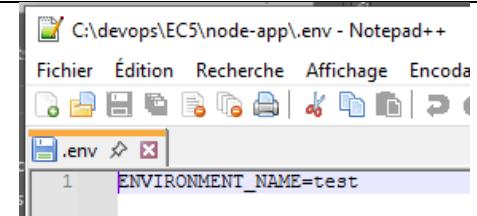
Workspaces

Workspaces	Run Status	Repo
PROD Default Project	✓ Applied	fabricе-git-hub/ec5
TEST Default Project	✓ Applied	fabricе-git-hub/ec5

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

4. CREATION DE L'APPLICATION NODE.JS

4.1. INITIALISATION

Création dossier node-app	
npm init -y	
Initialise un fichier package.json avec des valeurs par défaut. Ce fichier est essentiel pour : <ul style="list-style-type: none">• Gérer les dépendances (npm install)• Définir des scripts (ex: npm start, npm test)• Publier sur npm	
npm install express dotenv <ul style="list-style-type: none">• express : framework pour créer des applications web et des API en Node.js.• dotenv : Permet de charger des variables d'environnement à partir d'un fichier .env.	Ajoute ces deux paquets dans le dossier node_modules/. Met à jour le fichier package.json en ajoutant les deux dépendances.
Création fichier .env ENVIRONMENT_NAME=test ** Ne sera pas versionné ** mais pour effectuer un test en local avant commit sur Github et déploiement via TerraformCloud	

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

4.2. CREATION DU FICHIER "INDEX.JS"

*C:\devops\EC5\node-app\index.js - Notepad++

```

Fichier Édition Recherche Affichage Encodage Langage Paramètres Outils Macro Exécution Modules d'e
[File] [New] [Open] [Save] [Save As] [Print] [Find] [Replace] [Go To] [Search] [Settings] [Tools] [Macro] [Execution] [Modules]
.env index.js ✎ ✎
1  require('dotenv').config();
2  const express = require('express');
3  const app = express();
4  const port = 3000;
5
6  app.get('/', (req, res) => {
7      res.send(`Bienvenue dans l'Environnement de: ${process.env.ENVIRONMENT_NAME}`);
8  });
9
10 app.listen(port, () => {
11     console.log(`Server running on port ${port}`);
12 })

```

require('dotenv').config()	Permet de lire les variables d'environnement définies dans un fichier .env
const express = require('express')	Importe le module Express : framework Node.js utilisé pour créer des serveurs web facilement.
const port = 3000	Définit le port sur lequel le serveur va écouter
app.get('/', (req, res) => { res.send(`Bienvenue dans l'Environnement de: \${process.env.ENVIRONMENT_NAME}`); }	Créé une route GET pour l'URL / Quand on accède à l'URL, le serveur répond avec un message contenant la valeur de la variable d'environnement ENVIRONMENT_NAME
app.listen(port, () => { console.log(`Server running on port \${port}`); })	Démarre le serveur Express sur le port spécifié (3000).

4.3. TEST LOCAL

node index.js	\$ node index.js Server running on port 3000
Ouvrir le navigateur et lancer : http://localhost:3000/	<p>Bienvenue dans l'Environnement de: test</p>

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

5. CREATION DES SCRIPTS TERRAFORM

5.1. SCRIPT MAINT.TF ([EN ANNEXE](#))

Pour cette évaluation le script sera généré en un seul fichier **maint.tf** qui contiendra :

Un VPC	<pre># 1. VPC You, 10 hours ago 1 author (You) resource "aws_vpc" "main" { cidr_block = "10.0.0.0/16" You, 10 hours ago 1 author (You) tags = { Name = "main_vpc" } }</pre>
Une Internet Gateway	<pre># 2. Internet Gateway You, 10 hours ago 1 author (You) resource "aws_internet_gateway" "gw" { vpc_id = aws_vpc.main.id You, 10 hours ago 1 author (You) tags = { Name = "main_igw" } }</pre>
Un Subnet	<pre># 3. Subnet public You, 5 hours ago 1 author (You) resource "aws_subnet" "public" { vpc_id = aws_vpc.main.id cidr_block = "10.0.1.0/24" availability_zone = "us-east-1a" You, 10 hours ago 1 author (You) tags = { Name = "public_subnet" } }</pre>

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

Une Table de routage (associée au subnet)	<pre># 4. Table de routage You, 10 hours ago 1 author (You) resource "aws_route_table" "public" { vpc_id = aws_vpc.main.id You, 10 hours ago 1 author (You) route { cidr_block = "0.0.0.0/0" gateway_id = aws_internet_gateway.gw.id } You, 10 hours ago 1 author (You) tags = { Name = "public_rt" } } # 5. Association de la route avec le subnet You, 10 hours ago 1 author (You) resource "aws_route_table_association" "public_assoc" { subnet_id = aws_subnet.public.id route_table_id = aws_route_table.public.id }</pre>
Un Groupe de Sécurité (avec autorisation des ports 80 et 22 en entrée)	<pre># 6. Security group (SSH_HTTP) You, 5 hours ago 1 author (You) resource "aws_security_group" "ssh_http" { name = "allow_ssh_http" vpc_id = aws_vpc.main.id You, 10 hours ago 1 author (You) ingress { from_port = 22 to_port = 22 protocol = "tcp" cidr_blocks = ["0.0.0.0/0"] } You, 5 hours ago 1 author (You) ingress { from_port = 80 to_port = 80 protocol = "tcp" cidr_blocks = ["0.0.0.0/0"] } You, 10 hours ago 1 author (You) egress { from_port = 0 to_port = 0 protocol = "-1" cidr_blocks = ["0.0.0.0/0"] } tags = { Name = "allow_ssh_http" } }</pre>

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

Une Instance EC2 (linux Debian 11)

Au démarrage de l'instance un script sera lancé pour installation d'Ansible + Git et lancement du [playbook](#) (installation node.js + nginx)

```
user_data = file("install-ansible.sh")
```

```
# 7. Instance EC2
You, 5 hours ago | 1 author (You)
resource "aws_instance" "web" {
  ami           = "ami-000a08b963606bb82" # debian-11-amd64-2
  instance_type = "t2.micro"
  key_name      = var.key_name
  subnet_id     = aws_subnet.public.id
  vpc_security_group_ids = [aws_security_group.ssh_http.id]
  associate_public_ip_address = true
  user_data = file("install-ansible.sh")

You, 5 hours ago | 1 author (You)
tags = {
  Name = "web-${var.ENVIRONMENT_NAME}"
}
}
```

L'IP Publique sera affichée en sortie du plan Terraform

```
# 8. Output IP publique
You, 10 hours ago | 1 author (You)
output "ip_publique_web" {
  value      = aws_instance.web.public_ip
  description = "Adresse IP publique de la VM"
}
```

5.2. DECLARATION DES VARIABLES

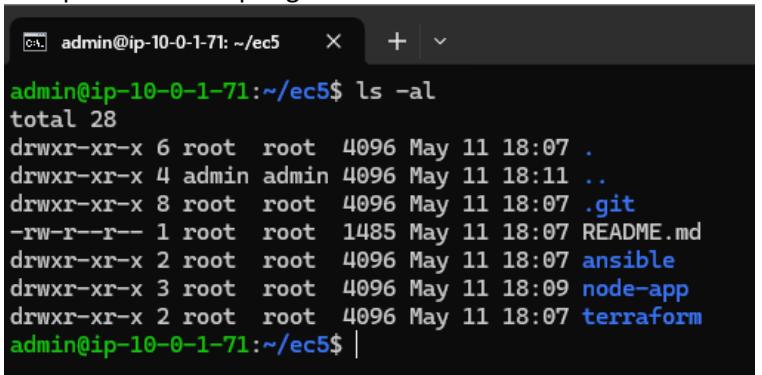
La valeur de ces variables sera renseignée via TerraformCloud (vu au paragraphe [Environnement TerraformCloud](#))

```
variables.tf > 📁 variable "AWS_SECRET_ACCESS_KEY"
You, 10 hours ago | 1 author (You)
1  ✓ variable "key_name" {
2    type = string
3    description = "Nom de la clé SSH"
4  }
5
You, 4 hours ago | 1 author (You)
6  ✓ variable "ENVIRONMENT_NAME" {
7    type = string
8    description = "Environnement (test ou prod)"
9  }
10
You, 4 hours ago | 1 author (You)
11 ✓ variable "AWS_ACCESS_KEY_ID" {
12   description = "Clé d'accès AWS"
13   type        = string
14   sensitive   = true
15 }
16
You, 4 hours ago | 1 author (You)
17 ✓ variable "AWS_SECRET_ACCESS_KEY" {
18   description = "Clé secrète AWS"
19   type        = string
20   sensitive   = true
21 }
```

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

5.3. SCRIPT BASH (INSTALLATION ANSIBLE)

```
install-ansible.sh
You, 5 hours ago | 1 author (You)
1 #!/bin/bash      You, 10 hours ago • first commit
2 apt update -y
3 apt install -y git ansible
4 git clone https://github.com/fabrice-git-hub/ec5.git /home/admin/ec5
5 cd /home/admin/ec5/ansible
6 ENVIRONMENT_NAME=${TF_WORKSPACE}
7 ansible-playbook -i "localhost," -c local playbook.yml
```

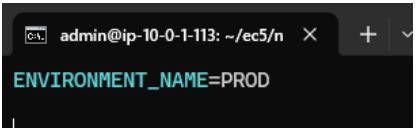
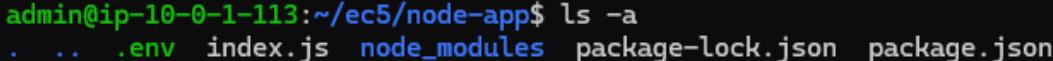
apt update -y apt install -y git ansible	Update des paquets et installation de git + ansible
git clone https://github.com/fabrice-git-hub/ec5.git /home/admin/ec5	Récupération du dépôt github EC5 
cd /home/admin/ec5/ansible ENVIRONMENT_NAME=\${TF_WORKSPACE} ansible-playbook -i "localhost," -c local playbook.yml	Création de la variable ENVIRONMENT_NAME en récupérant le nom de l'espace de travail TF_WORKSPACE est une variable d'environnement spéciale fournie automatiquement par Terraform. Elle contient le nom de l'espace de travail actuellement en cours d'exécution dans Terraform Cloud.

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

6. CREATION DU PLAYBOOK ANSIBLE ([EN ANNEXE](#))

<pre>- hosts: all become: true vars: environment_name: "{{ lookup('env', 'ENVIRONMENT_NAME') }}"</pre>	hosts: all : Le playbook s'applique à tous les hôtes cibles spécifiés dans l'inventaire. become: true : Exécute les tâches avec des privilèges root (sudo). vars : Ansible lit la variable d'environnement depuis la machine sur laquelle le playbook est exécuté et qui sera la valeur de ENVIRONMENT_NAME=\${TF_WORKSPACE}
<pre>tasks: - name: Update apt packages apt: update_cache: yes upgrade: dist</pre>	Met à jour la liste des paquets disponibles. Met à niveau tous les paquets vers la dernière version disponible
<pre>- name: Install Node.js and Nginx apt: name: - nodejs - npm - nginx state: present</pre>	Installe Node.js, npm et Nginx
<pre>- name: Configure Nginx copy: dest: /etc/nginx/sites-available/default content: server { listen 80; location / { proxy_pass http://localhost:3000; proxy_http_version 1.1; proxy_set_header Upgrade \$http_upgrade; proxy_set_header Connection 'upgrade'; proxy_set_header Host \$host; proxy_cache_bypass \$http_upgrade; } } notify: restart nginx</pre>	Écrit un fichier de configuration pour Nginx. Configure Nginx comme reverse proxy vers l'application Node.js qui tourne sur le port 3000 .

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

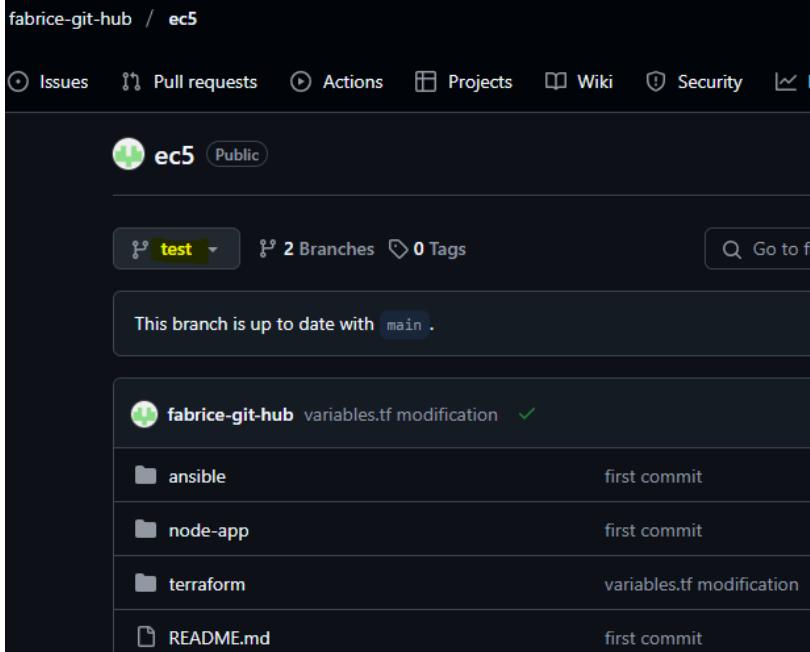
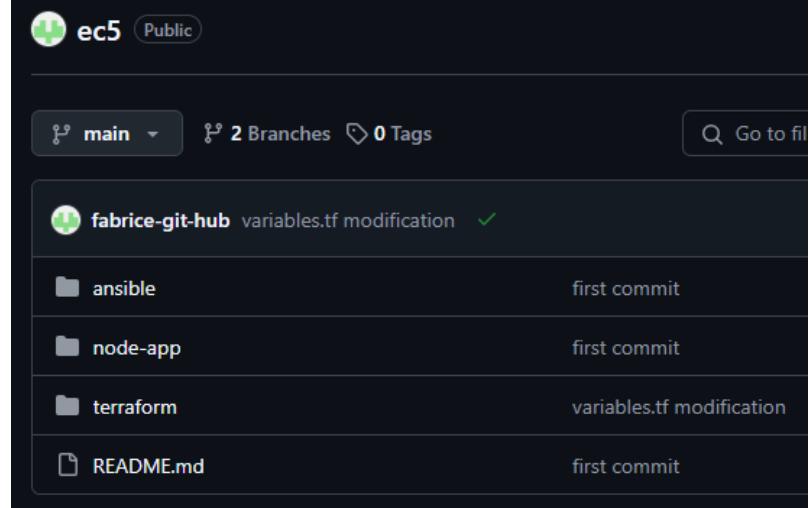
<pre>- name: Set ENV variable copy: dest: /home/admin/ec5/node-app/.env content: ENVIRONMENT_NAME={{ environment_name }}</pre>	Créé un fichier <code>.env</code> contenant une variable d'environnement pour l'application Node.js.
	
	
<pre>- name: Start Node.js app shell: cd /home/admin/ec5/node-app nohup node index.js &</pre>	Démarre l'application en arrière-plan avec <code>nohup</code> (évite qu'elle soit tuée après fermeture de session).
<pre> handlers: - name: restart nginx service: name: nginx state: restarted</pre>	Redémarre le service Nginx si la configuration change.

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

7. COMMIT DES BRANCHES TEST ET MAIN

git status	No commits yet Untracked files: (use "git add <file>..." to include in what will be committed) README.md ansible/ node-app/ terraform/
git add .	No commits yet Changes to be committed: (use "git rm --cached <file>..." to unstage) new file: README.md new file: ansible/playbook.yml new file: node-app/index.js new file: node-app/node_modules/.bin/mime new file: node-app/node_modules/.bin/mime.cmd new file: node-app/node_modules/.bin/mime.ps1 new file: node-app/node_modules/.package-lock.json new file: node-app/node_modules/accepts/HISTORY.md new file: node-app/node_modules/accepts/LICENSE new file: node-app/node_modules/accepts/README.md new file: node-app/node_modules/accepts/index.js new file: node-app/node_modules/accepts/package.json new file: node-app/node_modules/array-flatten/LICENSE new file: node-app/node_modules/array-flatten/README.md new file: node-app/node_modules/vary/HISTORY.md new file: node-app/node_modules/vary/LICENSE new file: node-app/node_modules/vary/README.md new file: node-app/node_modules/vary/index.js new file: node-app/node_modules/vary/package.json new file: node-app/package-lock.json new file: node-app/package.json new file: terraform/install-ansible.sh new file: terraform/main.tf new file: terraform/variables.tf
git commit	644 files changed, 69500 insertions(+) create mode 100644 README.md create mode 100644 ansible/playbook.yml create mode 100644 node-app/index.js create mode 100644 node-app/node_modules/.bin/mime create mode 100644 node-app/node_modules/.bin/mime.cmd create mode 100644 node-app/node_modules/.bin/mime.ps1 create mode 100644 node-app/node_modules/.package-lock.json create mode 100644 node-app/node_modules/accepts/HISTORY.md
git push -u origin test	Enumerating objects: 748, done. Counting objects: 100% (748/748), done. Delta compression using up to 4 threads Compressing objects: 100% (692/692), done. Writing objects: 100% (748/748), 740.30 KiB 925.00 KiB/s, done. Total 748 (delta 136), reused 0 (delta 0), pack-reused 0 (from 0) remote: Resolving deltas: 100% (136/136), done. To https://github.com/fabrice-git-hub/ec5.git

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

Vérification sur Github	
git checkout main git merge test git push origin main	<pre>ulukai@DESKTOP-8VK8LT7 MINGW64 /c/devops/EC5 (main) \$ git push -u origin main Enumerating objects: 748, done. Counting objects: 100% (748/748), done. Delta compression using up to 4 threads Compressing objects: 100% (692/692), done. Writing objects: 100% (748/748), 740.30 KiB 925.00 KiB/s, done. Total 748 (delta 136), reused 0 (delta 0), pack-reused 0 (from 0) remote: Resolving deltas: 100% (136/136), done. To https://github.com/fabrice-git-hub/ec5.git * [new branch] main -> main branch 'main' set up to track 'origin/main'.</pre>
Vérification sur Github	

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

8. LANCEMENT DES RUNS TERRAFORM

8.1. RUN TEST

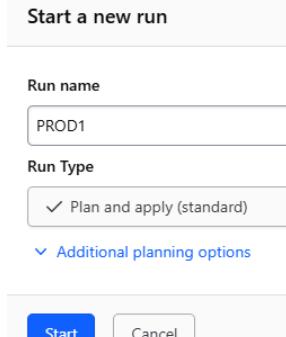
Un RUN sera lancé à chaque commit sur le repo ou pourra être lancé manuellement	<p>Start a new run</p> <p>Run name TEST1</p> <p>Run Type ✓ Plan and apply (standard)</p> <p>+ New run Start Cancel</p>
Terraform PLAN	<p>TEST</p> <p>ID: ws-D1VsVSMxjB1gKKzf </p> <p>Branche de TEST du repository Github ECS</p> <p>Running Resources 0 Tags 0 Terraform v1.11.4 Updated a few seconds</p> <p>TEST1</p> <p>Pending confirmation Re... Less than a minute</p> <p> fabrice-terraform triggered a run from UI a few seconds ago</p> <p>✓ Plan finished a few seconds ago</p> <p>Started a few seconds ago > Finished a few seconds ago</p> <p>+ 7 to create</p>
Terraform APPLY	<p>✓ Apply finished a few seconds ago Resources: 7 added, 0 changed, 0 destroyed</p> <p>Started a minute ago > Finished a few seconds ago</p> <p>+ 7 created</p> <p>Filter resources by address... Filter by action Terraform 1.11.4 Do</p> <p>> Diagnostics</p> <ul style="list-style-type: none"> > + aws aws_instance.web ✓ Created id=i-005eaba49ea2440ce > + aws aws_internet_gateway.gateway ✓ Created id=igw-07651725a52fd7356 > + aws aws_route_table_association.public_assoc ✓ Created id=rtbassoc-069ece6225951ceff > + aws aws_route_table.public ✓ Created id=rtb-0c4cae26e68349a2 > + aws aws_security_group_ssh.http ✓ Created id=sg-062445e98c1937e30 > + aws aws_subnet.public ✓ Created id=subnet-07da12396d7b2d0cc > + aws aws_vpc.main ✓ Created id=vpc-04887cd341340b386 <p>> Outputs 1 total ip_publique_web : "54.242.125.119"</p> <p>State versions created: org-node-app/TEST#sv-H8Zy5sCiyKzZKJmG (May 11, 2025 20:42:49 pm)</p>

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

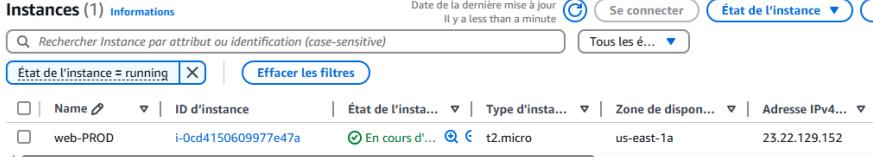
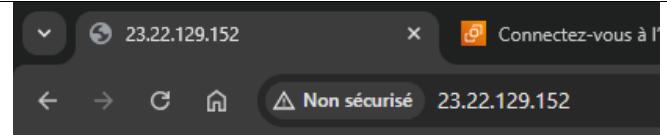
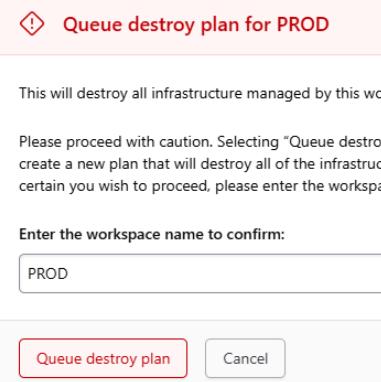
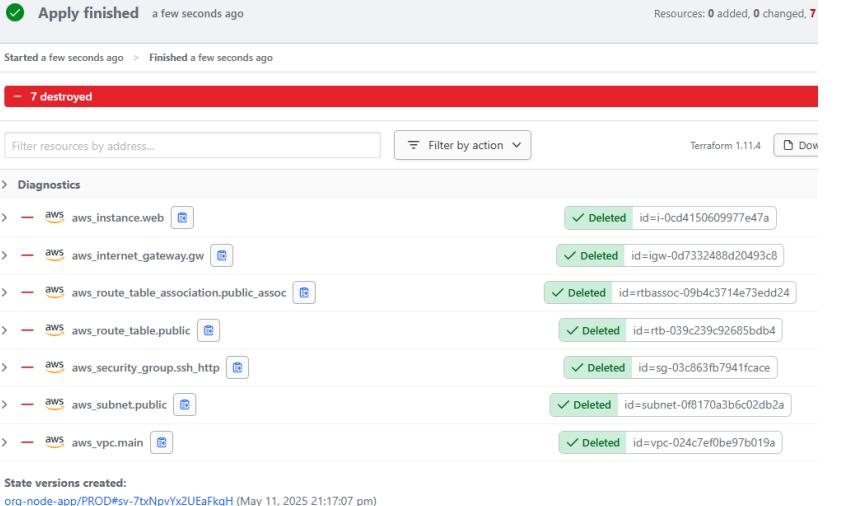
Vérification sur AWS de l'instance avec suffixe TEST	<p>Instances (1) Informations Date de la dernière mise à jour il y a 1 minute Se connecter État de l'instance Acchercher Instance par attribut ou identification (case-sensitive) Tous les é... État de l'instance = running Effacer les filtres</p> <table border="1"> <thead> <tr> <th>Name</th> <th>ID d'instance</th> <th>État de l'instance</th> <th>Type d'insta...</th> <th>Zone de dispon...</th> <th>Adresse IPv4...</th> </tr> </thead> <tbody> <tr> <td>web-TEST</td> <td>i-005eaba49ea2440ce</td> <td>En cours d'exéc... QC</td> <td>t2.micro</td> <td>us-east-1a</td> <td>54.242.125.119</td> </tr> </tbody> </table>	Name	ID d'instance	État de l'instance	Type d'insta...	Zone de dispon...	Adresse IPv4...	web-TEST	i-005eaba49ea2440ce	En cours d'exéc... QC	t2.micro	us-east-1a	54.242.125.119
Name	ID d'instance	État de l'instance	Type d'insta...	Zone de dispon...	Adresse IPv4...								
web-TEST	i-005eaba49ea2440ce	En cours d'exéc... QC	t2.micro	us-east-1a	54.242.125.119								
Accès au navigateur pour afficher le message de Bienvenue de node.js dans l'environnement de TEST	<p>54.242.125.119 Instances EC2 Non sécurisé 54.242.125.119</p> <p>Bienvenue dans l'Environnement de: TEST</p>												
Lancement RUN de Destruction	<p>Queue destroy plan for TEST</p> <p>This will destroy all infrastructure managed by this workspace.</p> <p>Please proceed with caution. Selecting "Queue destroy plan" creates a new plan that will destroy all of the infrastructure managed by this workspace. Once you select "Queue destroy plan", you cannot undo this action.</p> <p>Enter the workspace name to confirm:</p> <input type="text" value="TEST"/> <p>Queue destroy plan Cancel</p>												
Infrastructure AWS détruite	<p>Apply finished a few seconds ago Resources: 0 added, 0 changed</p> <p>Started a minute ago > Finished a few seconds ago</p> <p>- 7 destroyed</p> <p>Filter resources by address... Filter by action Terraform 1.11.4</p> <p>Diagnostics</p> <ul style="list-style-type: none"> aws aws_instance.web Deleted id=i-005eaba49ea2440ce aws aws_internet_gateway.gw Deleted id=igw-07651725a52fd7356 aws aws_route_table_association.public_assoc Deleted id=rtbassoc-069ece6225951ceff aws aws_route_table.public Deleted id=rtb-0c4c5ae26e68349a2 aws aws_security_group.ssh_http Deleted id=sg-062445e98c1937e30 aws aws_subnet.public Deleted id=subnet-07da12396d7b2d0cc aws aws_vpc.main Deleted id=vpc-04887cd341340b386 <p>State versions created: org-node-app/TEST#sv-kSad91U8gnnEgKHz (May 11, 2025 21:00:16 pm)</p>												

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

8.2. RUN PROD

Un RUN sera lancé à chaque commit sur le repo ou pourra être lancé manuellement	
Terraform PLAN	<p>PROD</p> <p>ID: ws-itU8goZDZh4yABz1 </p> <p>Branche de PROD du repository Github ECS</p> <p> Running  Resources 0  Tags 0  Terraform v1.11.4 </p> <p>PROD</p> <p>Pending confirmation Less than a minute</p> <p> fabrice-terraform triggered a run from UI a few seconds ago</p> <p> Plan finished a few seconds ago</p> <p>Started a few seconds ago > Finished a few seconds ago</p> <p>+ 7 to create</p>
Terraform APPLY Affichage de l'IP publique en sortie	<p> Apply finished a few seconds ago</p> <p>Resources: 7 added, 0 changed.</p> <p>Started a minute ago > Finished a few seconds ago</p> <p>+ 7 created</p> <p>Filter resources by address... Filter by action Terraform 1.11.4 </p> <p>> Diagnostics</p> <ul style="list-style-type: none"> >  aws aws_instance.web   id=i-0cd4150609977e47a >  aws aws_internet_gateway.gw   id=igw-0d7332488d20493c8 >  aws aws_route_table_association.public_assoc   id=rtbassoc-09b4c3714e73edd24 >  aws aws_route_table.public   id=rtb-039c239c92685bdb4 >  aws aws_security_group.ssh_http   id=sg-03c863fb7941fcace >  aws aws_subnet.public   id=subnet-0f8170a3b6c02db2a >  aws aws_vpc.main   id=vpc-024cef0be97b019a <p>< Outputs 1 total ip_publique_web : "23.22.129.152"</p> <p>State versions created: org-node-app/PROD#sv-oB3p1EcATeLR9pzR (May 11, 2025 21:05:52 pm)</p>

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

Vérification sur AWS de l'instance avec suffixe PROD	 <p>Instances (1) Informations Date de la dernière mise à jour Il y a less than a minute Se connecter État de l'instance ▾</p> <p>Rechercher Instance par attribut ou identification (case-sensitive) Tous les é... ▾</p> <p>État de l'instance = running Effacer les filtres</p> <table border="1"> <thead> <tr> <th>Name</th> <th>ID d'instance</th> <th>État de l'insta...</th> <th>Type d'insta...</th> <th>Zone de dispon...</th> <th>Adresse IPv4...</th> </tr> </thead> <tbody> <tr> <td>web-PROD</td> <td>i-0cd4150609977e47a</td> <td>En cours d'...</td> <td>t2.micro</td> <td>us-east-1a</td> <td>23.22.129.152</td> </tr> </tbody> </table>	Name	ID d'instance	État de l'insta...	Type d'insta...	Zone de dispon...	Adresse IPv4...	web-PROD	i-0cd4150609977e47a	En cours d'...	t2.micro	us-east-1a	23.22.129.152
Name	ID d'instance	État de l'insta...	Type d'insta...	Zone de dispon...	Adresse IPv4...								
web-PROD	i-0cd4150609977e47a	En cours d'...	t2.micro	us-east-1a	23.22.129.152								
Accès au navigateur pour afficher le message de Bienvenue node.js dans l'environnement de PROD	 <p>23.22.129.152 Connectez-vous à l'application</p> <p>Non sécurisé 23.22.129.152</p> <p>Bienvenue dans l'Environnement de: PROD</p>												
Lancement RUN de Destruction	 <p>Queue destroy plan for PROD</p> <p>This will destroy all infrastructure managed by this workspace.</p> <p>Please proceed with caution. Selecting "Queue destroy plan" will create a new plan that will destroy all of the infrastructure in this workspace. If you are certain you wish to proceed, please enter the workspace name to confirm:</p> <p>Enter the workspace name to confirm:</p> <p>PROD</p> <p>Queue destroy plan Cancel</p>												
Infrastructure AWS détruite	 <p>Apply finished a few seconds ago Resources: 0 added, 0 changed, 7 destroyed</p> <p>Started a few seconds ago > Finished a few seconds ago</p> <p>- 7 destroyed</p> <p>Filter resources by address... Filter by action ▾ Terraform 1.11.4 Download</p> <p>Diagnostics</p> <ul style="list-style-type: none"> aws aws_instance.web Deleted id=i-0cd4150609977e47a aws aws_internet_gateway.gw Deleted id=igw-0d7332488d20493c8 aws aws_route_table_association.public_assoc Deleted id=rtbassoc-09b4c3714e73edd24 aws aws_route_table.public Deleted id=rtb-039c239c92685bdb4 aws aws_security_group.ssh_http Deleted id=sg-03c863fb7941fcace aws aws_subnet.public Deleted id=subnet-0f8170a3b6c02db2a aws aws_vpc.main Deleted id=vpc-024c7ef0be97b019a <p>State versions created: org-node-app/PROD#sv-7txNpvYx2UEaFkqH (May 11, 2025 21:17:07 pm)</p>												

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

9. ANNEXE

9.1. SCRIPT TERRAFORM

```

1. terraform {
2.   required_providers {
3.     aws = {
4.       source  = "hashicorp/aws"
5.       version = "5.95.0" #
6.     }
7.   }
8.   required_version = "1.11.4"
9. }
10.provider "aws" {
11.   region      = "us-east-1"
12. }
13.
14.# 1. VPC
15.resource "aws_vpc" "main" {
16.   cidr_block          = "10.0.0.0/16"
17.   tags = {
18.     Name = "main_vpc"
19.   }
20. }
21.
22.# 2. Internet Gateway
23.resource "aws_internet_gateway" "gw" {
24.   vpc_id = aws_vpc.main.id
25.   tags = {
26.     Name = "main_igw"
27.   }
28. }
29.
30.# 3. Subnet public
31.resource "aws_subnet" "public" {
32.   vpc_id           = aws_vpc.main.id
33.   cidr_block       = "10.0.1.0/24"
34.   availability_zone = "us-east-1a"
35.   tags = {
36.     Name = "public_subnet"
37.   }
38. }
39.

```

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

```

40. # 4. Table de routage
41. resource "aws_route_table" "public" {
42.   vpc_id = aws_vpc.main.id
43.
44.   route {
45.     cidr_block = "0.0.0.0/0"
46.     gateway_id = aws_internet_gateway.gw.id
47.   }
48.
49.   tags = {
50.     Name = "public_rt"
51.   }
52. }
53.
54. # 5. Association de la route avec le subnet
55. resource "aws_route_table_association" "public_assoc" {
56.   subnet_id      = aws_subnet.public.id
57.   route_table_id = aws_route_table.public.id
58. }
59.
60. # 6. Security group (SSH_HTTP)
61. resource "aws_security_group" "ssh_http" {
62.   name          = "allow_ssh_http"
63.   vpc_id        = aws_vpc.main.id
64.
65.   ingress {
66.     from_port    = 22
67.     to_port      = 22
68.     protocol     = "tcp"
69.     cidr_blocks = ["0.0.0.0/0"]
70.   }
71.   ingress {
72.     from_port    = 80
73.     to_port      = 80
74.     protocol     = "tcp"
75.     cidr_blocks = ["0.0.0.0/0"]
76.   }
77.
78.   egress {
79.     from_port    = 0
80.     to_port      = 0
81.     protocol     = "-1"
82.     cidr_blocks = ["0.0.0.0/0"]
83.   }
84.
```

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

```

85.   tags = {
86.     Name = "allow_ssh_http"
87.   }
88. }
89.
90.# 7. Instance EC2
91.resource "aws_instance" "web" {
92.   ami           = "ami-000a08b963606bb82" # debian-11-amd64-20250402-2070 sur eu-
93.   instance_type      = "t2.micro"
94.   key_name          = var.key_name
95.   subnet_id         = aws_subnet.public.id
96.   vpc_security_group_ids = [aws_security_group.ssh_http.id]
97.   associate_public_ip_address = true
98.   user_data = file("install-ansible.sh")
99.
100.  tags = {
101.    Name = "web-${var.ENVIRONMENT_NAME}"
102.  }
103. }
104.
105. # 8. Output IP publique
106. output "ip_publique_web" {
107.   value      = aws_instance.web.public_ip
108.   description = "Adresse IP publique de la VM"
109. }
110.

```

9.2. SCRIPT ANSIBLE

```

---
- hosts: all
  become: true
  vars:
    environment_name: "{{ lookup('env','ENVIRONMENT_NAME') }}"
  tasks:
    - name: Update apt packages
      apt:
        update_cache: yes
        upgrade: dist
    - name: Install Node.js and Nginx

```

Evaluation	Parcours	ADMINISTRATEUR SYSTEME DEVOPS Préparer un environnement de test	Auteur	FT
ECF5	Titre		Du	10/11/2025

```

apt:
  name:
    - nodejs
    - npm
    - nginx
  state: present

  - name: Configure Nginx
  copy:
    dest: /etc/nginx/sites-available/default
    content: |
      server {
        listen 80;
        location / {
          proxy_pass http://localhost:3000;
          proxy_http_version 1.1;
          proxy_set_header Upgrade $http_upgrade;
          proxy_set_header Connection 'upgrade';
          proxy_set_header Host $host;
          proxy_cache_bypass $http_upgrade;
        }
      }
  notify: restart nginx

  - name: Set ENV variable
  copy:
    dest: /home/admin/ec5/node-app/.env
    content: |
      ENVIRONMENT_NAME={{ environment_name }}

  - name: Start Node.js app
  shell: |
    cd /home/admin/ec5/node-app
    nohup node index.js &

handlers:
  - name: restart nginx
  service:
    name: nginx
    state: restarted

```