

Ingénieur Full Stack / DevOps

5 années d'expérience

COMPÉTENCES PRINCIPALES

Cloud : Terraform, Terragrunt, Docker, sonarqube , Kubernetes, AWS, Azure

CI/CD : GitLab CI/CD, Jenkins , Github Actions

OS : Linux , Windows

Outils projet : JIRA , GIT, ASANA

Languages : NodeJS, Python, HCL, Javascript , Typescript , Bash Linux , Shell

Framework : Angular, RxJS, Express,

Base de données : SQL (PostgreSQL, MySQL), NoSQL (MongoDB, AWS DynamoDB)

Monitoring : Prometheus , Grafana , Alert Manager, Datadog , ELK

Methodologies : Scrum , Kanban , Safe

Autre : Git , Postman

Langues : Anglais , Français , Espagnol

DIPLÔME & FORMATIONS

- **2023** : Terraform
- **2020** : AWS Developer - Associate
- **2018** : Diplôme d'ingénieur en Informatique (**ESPRIMS**)

CERTIFICATIONS

- **2022** : Containers & Kubernetes Essentials
- **2022** : Docker Essentials

EXPERIENCES PROFESSIONNELLES

01/2023 – 03/2024 : Développeur FullStack / DevOps AWS @BPIFRANCE

Bpifrance est une banque publique d'investissement française, ayant pour mission le financement et le

MALIKA WANNASI
malika.wannasi@outlook.fr
+33 6 05 71 89 06
78140 Vélizy

développement des entreprises. Elle est née à la fin de l'année 2012 du regroupement d'Oséo, de CDC Entreprises, du Fonds stratégique d'investissement et du FSI Régions

Organization : 4 dev, 1 PO , 1 tech lead , 1 scrum master

Méthodologie de travail: Safe

Réalisation:

- Développement Back/Front
- Développement MicroServices; Api, Frontend
- Stratégie de test unitaires, fonctionnels/conformité
- Création Modules terraform
- Création Chaînes CI/CD en mode Gitops, déploiement sur EKS, AWS
- Automatisation infrastructure dans le cloud AWS, ou Atlas/Elastic
- Migration des bases de données avec Mongoimport/ mongo miroir
- Ajouter les permissions sur les rôles des EC2
- Documentation , support aux équipe

Technologies utilisées: Terraform, Kafka, MongoDB, Gitlab runners, Gitlab CI, AWS, vue Js , node js , postman, kubernetes , ELK , Kibana , Elasticsearch , Datadog , ansible , jenkis , PostgreSQL, EC2, Vault .

07/2022 – 12/2022 : Ingénieur DevOps @Crédit Agricole

Le Crédit agricole est une banque française proposant des services financiers à l'échelle internationale, le groupe possédant le plus grand réseau de banques coopératives et mutualistes au monde, dont trente-neuf caisses régionales de Crédit agricole en France.

- Sauvegarde et restauration de base de données MongoDB avec mongodump et mongorestore
- Automatisation du processus de sauvegarde en utilisant Gitlab Schedule et Terraform
- Déploiement d'environnements DEV, UAT et PROD avec Terraform et AWS
- Simplification du processus de déploiement avec un pipeline de lancement de tâches
- Automatisation du déploiement sur l'environnement DEV en utilisant Gitlab CI
- Configuration de Gitlab Runner avec l'exécuteur Docker
- Migration de l'application backend de Spring Boot vers Node.js
- Migration de la base de données de MySQL vers MongoDB à l'aide de scripts Python
- Gestion de l'infrastructure avec Terraform, AWS et Terragrunt.

Technologies utilisées: Terragrunt, Terraform, Docker, MongoDB, Gitlab runners, Gitlab CI, AWS, Mysql, NodeJs.

07/2020 – 06/2022: Développeur FullStack/DevOps AWS- Azure@Charlize

Charliz est une société éditrice de logiciels spécialisée dans le développement d'applications web et mobile, offrant des solutions innovantes adaptées à divers secteurs d'activité tels que la banque et l'enseignement.

PROJET :

Contexte : Conception, développement et mise en place d'une solution web, pour la gestion des entreprises, de la comptabilité et de leurs employés.

Organization : 7 dev, 1 PO , 1 tech lead

Méthodologie de travail: Kanban

Réalisation:

- Analyse et spécification des besoins
- Gestion des rôles dans la plate-forme .
- Utilisation d'une architecture en microservice.
- Migration de l'application backend de Spring Boot vers Node.js
- Migration de l'application Frontend de java vers Angular
- Authentification sociale avec AWS Cognito.
- Authentification sociale avec Azure Active Directory B2C (Azure AD B2C).
- Configuration d'un environnement docker en local pour la base de données mongo et mongoDB client en utilisant terraform.
- Utilisation d'une base de données non relationnelle avec mongodb.
- Création des API REST en utilisant Node.js.
- Utilisation d'un contrat Swagger pour générer la documentation pour les API REST.
- Création des services angular pour consommer les REST API exposés dans la partie backend.
- Implémentation des stages CI/CD pour la partie déploiement et les tests.
- Maintenir et faire évoluer la chaîne CI/CD en GitLab
- Gestion multilingue avec ngx-translate
- Importation et exportation Fichier excel dans la plateforme
- Créer partie de statistique à partir des données stockées dans Mongodb et utiliser agrégation pipeline
- Signature électronique des contrats.
- Génère des factures au format PDF à partir des données stockées dans la base de données.

- Paiement des factures en ligne.
- Envoyer des e-mails par simple clic.
- Tester la qualité du code avec SonarQube

Technologies utilisées: Terragrunt, Terraform, Docker, MongoDB, Gitlab runners,

Gitlab CI, AWS, Azure, angular , node js , postman , SonarQube.

PROJET :

Contexte : Conception, développement et mise en place d'une jardin d'enfant .

Organization : 15 dev, 1 PO , 1 tech lead

Méthodologie de travail: Scrum

Réalisation:

- Création et développement de la partie backend avec Node.js et tests d'API avec Postman
- Conception de la base de données MongoDB pour l'application
- Implémentation du front-end avec Angular 13
- Gestion multilingue avec ngx-translate
- Mise en place de la chaîne CI/CD avec Gitlab CI pour automatiser le processus de validation, de construction et de déploiement
- Accompagnement des nouveaux arrivants dans l'architecture de l'infrastructure et les outils utilisés : Terraform, AWS, CI/CD

Standardisation du processus de création de projet Gitlab avec Terraform.

Technologies utilisées: Terragrunt, Terraform, Docker, MongoDB, Gitlab runners,

Gitlab CI, AWS, angular , node js , postman.

PROJET : Un provider docker:

- Le provisioning de la création de nos conteneurs de base de données(Mongo/ MySql) & gitlab runners dans le cloud privé.
- La création de nos conteneurs dans l'environnement local Mongo et Mongo-Client.
- Mise en place d'une solution de monitoring terraformé en utilisant prometheus, grafana et alert manager

PROJET : Mise en place d'une solution IAC de déploiement pour les environnements DEV , UAT , et PROD en utilisant un seul compte master AWS et une organisation pour chaque projet

- Développement de backend Node.js avec AWS Lambda
- Gestion des comptes IAM AWS et des privilèges d'accès avec Terraform et Gitlab
- Gestion des environnements DEV, UAT et PROD avec un compte master et une organisation par environnement

- Utilisation de AWS API Gateway pour la gestion des URLs d'API
- Ajout d'un domaine personnalisé et installation de certificats SSL pour l'API et la partie

front-end dans le cloud AWS.

Technologies utilisées: Terragrunt, Terraform, Docker, MongoDB, Gitlab runners, Gitlab CI, AWS

PROJET : Mise en place d'une solution IAC de déploiement pour les environnements DEV , UAT , et PROD en utilisant terraform et Azure

- Développement de fonctions serverless avec Azure Functions.
- Gestion des comptes et des privilèges d'accès avec Azure Active Directory (Azure AD) et Azure Resource Manager (ARM) via Terraform et Azure DevOps.
- Gestion des environnements DEV, UAT et PROD avec Azure DevOps et Azure Resource Manager (ARM) pour la gestion des groupes de ressources.
- Utilisation d'Azure API Management pour la gestion des URLs d'API.
- Configuration d'un domaine personnalisé et installation de certificats SSL avec Azure App Service pour l'hébergement de l'API et des parties front-end dans Azure.

Technologies utilisées: Terragrunt, Terraform, Docker, MongoDB, Gitlab runners, Gitlab CI, Azure.

PROJET : Migration des données de la base de données MySQL vers la base de données Mongodb

- Développez scripts python qui s'exécutent si le script s'exécute en localhost sinon il s'exécute dans un seul processus lorsque le runner lance ce script.
- Développer un script terraform avec le provider Gitlab pour créer un scheduler qui se déclenche tous les 3 jours pour synchroniser les données et exécuter les scripts python de migration.

Technologies utilisées: SQL, Python, PyMongo , Gitlab, Terraform, Terragrunt.

PROJET : Configuration de l'infrastructure AWS et Docker avec Terraform pour le déploiement d'une application web.

- Écrire un script Terraform qui configure l'infrastructure AWS pour le projet, incluant la création d'un Bucket S3 pour stocker les fichiers de build du front-end, d'un Lambda pour le fichier zip du build du back-end, d'une API Gateway pour les requêtes Rest API définies dans le fichier swagger, et d'un Cognito pour la gestion de l'authentification.
- Écrire un script Terraform qui configure l'infrastructure en localhost avec Docker, en créant deux conteneurs pour Mongo et MongoClient.
- Rédiger le fichier gitlab-ci.yml pour la configuration des pipelines de build et de tag du back-end et du front-end. Créer une image Docker contenant toutes les commandes nécessaires pour exécuter les scripts du pipeline sur un runner GitLab.

Technologies utilisées: Terragrunt, Terraform, Docker, MongoDB, Gitlab runners, Gitlab CI, AWS.

PROJET : FinOps:

- Utilisation de AWS Cost Manager pour visualiser les coût AWS par projet et par services avec les estimations
- Minimisations des coûts avec désactivation des logs non utiles sur les environnement DEV, UAT et PROD pour minimiser les coûts de AWS Cloud Watch.

Technologie utilisée: S3, Lambda, API Gateway, CloudFront, Cognito, IAM, Route53, Cost Explorer, AWS Cloud Watch, Angular, Nodejs, gitlab, CI/CD, GITOPS, Swagger

10/2018 – 01/2020: Développeur FullStack @Allence

Allence Tunisie est une société de conseil en système d'information disposant d'une forte expertise dans le domaine du développement.

PROJET : Conception, développement et Implémentation d'une solution web pour la gestion vente aux enchères en ligne

Réalisations :

- Analyse et spécification des besoins
- Gestion des rôles dans la plate-forme.
- Utilisation d'une architecture en microservice.
- Mise en place d'une base de données NoSql Postgres
- Développer des modules en backend avec Node.js (client, commande, achat, vente, boutique, ...)
- Assurer l'authentification en utilisant JWT
- Développement des fonctionnalités au niveau du frontEnd
- Développement des interfaces avec angular Material UI
- Intégration des tests unitaire au niveau du front end avec JEST

Environnement : NodeJS, ExpressJS, Angular, Postges, GitLab, Docker