Etat de l’art

# Introduction :

Le but de cet état de l’art est de faire l’inventaire de l’état de la recherche sur des projet de CaaS ou plus généralement sur l’utilisation de Docker. Dans une seconde partie nous regarderons quelques solutions existantes se rapprochant de nos attentes.

# État de la recherche :

Après une recherche sur des moteurs de recherche dédiés j’ai trouvé cinq papiers répondant à mes critères (cf dossier recherche). Cependant je n’ai majoritairement trouvé que des papiers de recherche datant de 2016 et un papier de 2019.

Ces papiers restent très évasifs quant à la mise en place d’un service de déploiement de conteneur mais esquisse son architecture et ses fonctionnalités principales. On peut cependant retenir qu’il est important dans une optique de répondre à la demande des entreprises mettant en place l’intégration continue un environnement de test avant mise en production permettant de prévenir de possible défaillance et assurer la continuité du service. Un autre point important est la sécurité. En effet dans le cas d’un service sur le cloud il est important que les données et projets soient isolés et ne puissent être compromis ou volés.

# Analyse de l’existant :

Pour cette étude nous avons retenu 6 services que sont AWS, Google cloud, Microsoft Azur, Heroku, Platform.sh et Clever cloud.

Trois de ces services (AWS, Google cloud et Microsoft azure) sont des services de cloud généralistes dont une partie de leurs offres est dédié à la conteneurisation. Ces trois services seront analysés en même temps car ayant des caractéristiques et offre similaire.

Les trois autres services sont de type PaaS proposant le déploiement de projet.

Les descriptions que je fais par la suite ne sont que la synthèse de mon expérience et de mon ressenti lors de l’utilisation des différents services et peuvent ne pas complètement correspondre à la réalité n’ayant pas une connaissance complète des possibilités offertes.

## Service de cloud

Les services de cloud que sont AWS Google Cloud et Microsoft Azure pour les plus connues, ont un fonctionnement similaire. Ils proposent un service de déploiement de conteneur avec généralement une offre d’orchestration basé sur Kubernetes. Néanmoins il semble que pour une partie des entreprises il soit préféré la location de VM sur lequel est mit un moteur Docker pour permettre une gestion plus pousser.

Au niveau du monitoring des conteneurs, les services de cloud offrent des moyens de surveillance standard à savoir la charge CPU et réseaux mais reste peu accessible et difficilement.

Du côté de l’utilisation les services cloud propose généralement une CLI à installer sur son poste de travail afin de ne pas avoir à passer par l’interface web. Certains services comme google cloud propose même une « console web » permettant de réaliser les actions comme avec l’interface CLI mais directement sur une VM pour ne travailler que sur le cloud.

## Clever Cloud

Clever cloud est un service PaaS qui permet de déployer des projets git dans le cloud. C’est le service qui se rapproche le plus de mes attentes. Le site propose différents environnements spécifiques pour différents langages et notamment pour docker.

Pour le déploiement il suffit de renseigner son identifiant GitHub puis de choisir le projet à déployer.

Seul les dockerfile sont supportés et on ne peut déployer qu’un conteneur par projet, ce qui ne permet pas le déploiement de projet de type micro-services.

Pour la partie logs, on a accès aux logs de notre conteneur et ils sont découpés par session de déploiement. Pour les métriques on a accès aux principaux que sont la consommation CPU et RAM et l’utilisation du réseau.

Le site est plutôt simple d’utilisation et intuitif. Lors du déploiement peu d’étapes sont nécessaires si on part d’un répertoire git. De plus, Clever cloud propose une application CLI pour gérer ces projets.

## Heroku et Platform.sh

Les deux derniers services que sont Heroku et Platform.sh ne remplisse pas complètement les critères que je me suis fixé. En effet malgré la possibilité de pouvoir construire un projet à partir d’un répertoire git, il n’est pas possible de prendre comme environnement docker.