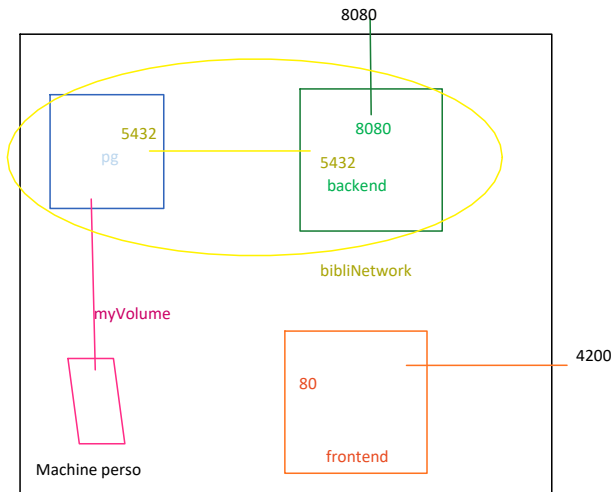


## Séance 4

mercredi 30 novembre 2022 07:58



On commence la séance en mettant à jours notre docker-compose.yml

On ajoute pour cela :

- Un indication de build :  
`build: ./back-main`
- Une dépendance :  
`depends_on: -postgres`

On rebuild le tout : **docker compose build** | en se plaçant dans la racine commune au front et au back

Afin de résoudre le problème du front qui ne démarre pas, et sans log, nous apportons quelques modifications au dockerfile et au endpoint

On a créé dans un second temps un autre fichier yaml.

Il est plus respectueux des règles de bienfaisance que l'initial et est donc voué à rester.

On a par exemple :

- Supprimer le nommage manuel des conteneurs (alias)
- Eviter la répétition des variables d'environnement (via un fichier .env)
- Ajouter une condition de redémarrage du conteneur selon l'état de son service (healthcheck)

**docker-compose -f docker-compose2.yml down --volume** : pour supprimer tous les conteneurs du docker-compose ainsi que le volume !

Maintenant que nos images sont bonnes, nous allons les publier !

Pour cela, nous commençons par les renommer : **docker tag 921c20b7806e public.ecr.aws/u1k0q0r7/back:juliette**

Puis on les push : **docker push public.ecr.aws/u1k0q0r7/back:juliette**

On peut ensuite récupérer l'image d'un de nos copains : **docker pull public.ecr.aws/u1k0q0r7/back:axel**

Pour l'utiliser, on appelle dans un docker-compose3.yml les images d'Axel avant de up ce yaml : **docker-compose -f docker-compose3.yml up -d**

**backend:**

`image: public.ecr.aws/u1k0q0r7/back:axel`

`build: ./back`