

******* Pour tous les exercices, vous travaillez en binôme *******

Exercice 1 – Analyse et implémentation du système de taxis.

Reprenons le mini système de taxis qui on été modélisé avec les services :

https://depinfo.u-cergy.fr/~tliu/ens/adm/taxi_soa.png

Supposons maintenant que chaque service se concrétise par un composant déployable. **Analyser** les dépendances entre les composants pour vérifier si cela respecte les différents principes vus en cours.

Implémentation de la solution en Java

On se base sur la solution :

https://depinfo.u-cergy.fr/~tliu/ens/adm/taxi_soa_cd.png

On n'a pas besoin de faire de fichiers jar pour différents composants. **Idées de la mise en œuvre :**

- ✓ Pour IHM : vous faites une IHM textuelle (console) simplifiée.
- ✓ Pour chaque composant (service), c'est un package dans lequel il y a une seule classe « interface » publique, les autres classes locales ne sont pas publiques (ne pas mettre le mot clé « public » devant ces classes)
- ✓ Obtention de l'objet de chaque interface peut se faire par deux solutions possibles :
 - Soit on utilise une simple « factory » pour créer l'objet de classe concrète qui implémente « interface », sans rendre cette classe concrète publique, la classe « factory » sera publique
 - Soit on rend cette classe concrète publique mais de préférence sous forme de « singleton »
- ✓ Pour l'interface « Taxi », on peut avoir une classe publique qui implémente
- ✓ Au sein de chaque composant, vous proposez une implémentation locale simple et logique en utilisant les éléments de programmation appropriés (List, Map, etc.), mais sans coder en dur.
- ✓ Vous créez un test dans le composant « gui » (ici, on ne fait que « ui » car IHM textuelle), pour tester votre programme réalisé