

	<p>Master Pro TAA/ 2014-2015</p>	
---	--------------------------------------	--

**TP A/TAA: Construire une application n-tiers à base du socle technologique
JPA, Jersey, easyrest**

Objectif :

L'objectif de ce TP est de prendre en main *un socle technologique* composé de JPA, JERSEY ou easyrest afin de construire une application web permettant la mise en place d'une application de covoiturage pour un évènement donné et une communauté donnée.

Cahier des charges de l'application.

Depuis un smartphone ou une application web, je dois pouvoir pour un évènement donné (date, lieu, heure) proposer un mécanisme de covoiturage automatique. Dès lors, plusieurs personnes vont proposer de prendre leur véhicule en précisant ces caractéristiques (nombre de places, ...). D'autres vont juste demander à être emmenées. En fonction de la localisation des différentes personnes, de leurs préférences et des voitures disponibles, le système compose automatiquement les équipages. Il doit être possible de noter ou laisser des commentaires à la fin sur cette composition d'équipage ou sur un équipage particulier.

L'objectif des TP est de réaliser en JEE, la partie backend serveur de cette application.



1/ Faites les modèles métier de votre application.

2/ Proposez une spécification du mapping OR. Construisez efficacement le lien entre ce modèle métier et une base de données relationnelle.

3/ Identifiez les différents services.

Regroupez ces services aux sein de ressources au sens (REST), faites les choix de mapping des services sur les opérations de base (GET, PUT, DELETE, POST)

Vous pouvez utiliser

<https://chrome.google.com/webstore/detail/dev-http-client/aejoelaoggembcahagimdiliamlcdmfm>
pour tester vos services.

4/ Réalisez l'application, un client lourd en mode texte est suffisant pour tester l'application mais n'hésitez pas à faire mieux.

Voici la liste des dépendances maven et des plugins utiles à votre application.

N'oubliez pas d'inclure un persistence.xml dans un répertoire META-INF dans vos ressources. Un template de projet est disponible [ici](#).

N'oubliez pas de lancer votre base de données afin de pouvoir vous y connecter.

```
<dependency>
  <groupId>com.sun.jersey</groupId>
  <artifactId>jersey-server</artifactId>
  <version>${jersey-version}</version>
</dependency>
<dependency>
  <groupId>com.sun.jersey</groupId>
  <artifactId>jersey-json</artifactId>
  <version>${jersey-version}</version>
</dependency>
<dependency>
  <groupId>org.hibernate</groupId>
  <artifactId>hibernate-entitymanager</artifactId>
  <version>4.1.7.Final</version>
</dependency>
<dependency>
  <groupId>org.hsqldb</groupId>
  <artifactId>hsqldb</artifactId>
  <version>2.2.8</version>
</dependency>
<dependency>
  <groupId>mysql</groupId>
  <artifactId>mysql-connector-java</artifactId>
  <version>5.1.21</version>
</dependency>
```

pour les plugins

```
<plugin>
  <groupId>org.apache.tomcat.maven</groupId>
```

```
<artifactId>tomcat7-maven-plugin</artifactId>
<version>2.1</version>
<configuration>
    <path>/</path>
</configuration>
</plugin>
```

Si certains sont un peu perdus, je vous ai mis sur share un projet vide configuré convenablement.

N'oubliez pas d'utiliser git ;)

Remarque

Il est important de bien comprendre le pourquoi de chaque action que vous effectuez dans ce TP. Si vous ne voyez pas l'intérêt de certaines de ces actions ou si vous trouvez certaines commandes magiques, n'hésitez pas à poser de nombreuses questions.

Documentation

Les transparents de cours évidemment ...