

- a) Comment le client appelle-t-il une ressource ? Illustrez votre réponse par l'appel de la liste des adhérents ?

Le client appelle et récupère une ressource via une requête HTTP. Dans le cadre d'un appel pour obtenir la liste des adhérents, le contrôleur va invoquer un appel JSON sur la classe Appel.java qui va elle-même appeler la classe WebTarget. Au final, c'est une requête HTTP qui est émise : <http://localhost:8080/mediatheque/Adherents> (GET).

- b) Dans le contrôleur du client, que contient la variable ressource ?

La variable ressource contient l'URI sur laquelle on va appeler le WebService. Autrement dit, l'URI où se trouve la ressource distante.

- c) Dans le web service quelle est la méthode qui transforme les données (collection, objet) sous forme de flux JSON ?

C'est la méthode « gson.toJson(mesAdherents) » qui permet de transformer une instance Java en un flux JSON. L'API de Google est capable de mapper chaque champ d'une instance Java en une représentation JSON.

- d) Quel est le rôle de la classe consommateur.java?

La classe Consommateur.java consomme le WebService proposé par le seueur via une URI. Dans le cas du projet, elle va créer un client HTTP et lui spécifier les informations en lien avec l'URI. Après quoi, elle va émettre la requête HTTP. A noter également que c'est un singleton car il ne faut qu'une seule instance !

- e) Quelle est le rôle de la classe WService.java ?

C'est la classe qui expose les fonctionnalités au travers d'une URI aux potentiels clients. Lorsqu'on ajoute une méthode dans la classe, on peut lui ajouter des annotations qui permettront de spécifier comment invoquer cette méthode via une requête HTTP.