

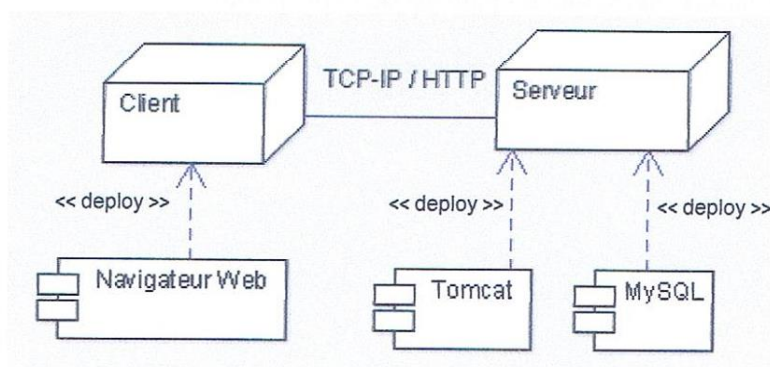
## Contrôle de Génie Logiciel 2

### Questions de Cours (6 Points)

1. Qu'est-ce qu'une ligne de produits logiciels ? Quel est son intérêt principal ? Donner un exemple de ligne de produits logiciels sous forme de diagramme FODA (Feature Oriented Domain Analysis).
2. Quel **énoncé est vrai** à propos des lignes de produit logiciels ?  
L'ingénierie des lignes de produits logiciels passe nécessairement par trois phases complémentaires, à savoir :
  - a) l'ingénierie du domaine, la gestion de la variabilité et l'ingénierie d'application.
  - b) l'ingénierie du domaine, la gestion de la Commonalité et l'ingénierie d'application.
  - c) l'ingénierie du domaine, la gestion des contraintes et l'ingénierie d'application.
  - d) Aucun de ces énoncés.
3. Quel **énoncé est vrai** à propos des styles architecturaux ?
  - a) **L'architecture Pipeline** est l'architecture la mieux adaptée pour modéliser les **systèmes interactifs**.
  - b) **FODA** (Feature Oriented Domain Analysis) est un **style architectural** pour les **applications de base de données**.
  - c) Aucun de ces énoncés.
4. Quel **énoncé est vrai** à propos du style architectural MVC (Model View Controller) ?  
MVC permet la séparation entre :
  - a) Les attributs et les méthodes des différentes classes modélisant le système concerné.
  - b) Les données, la présentation et les traitements du système concerné.
  - c) Les différents Packages du système concerné.

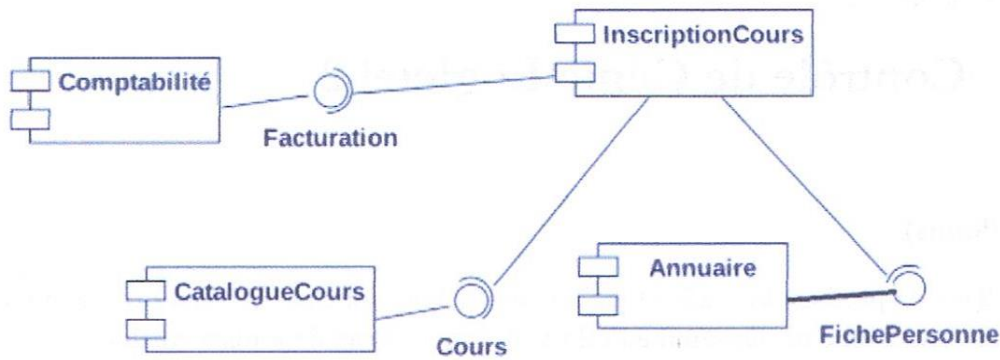
### Exercice 1 (6 Points): Les réponses doivent être détaillées.

1. Soit le diagramme UML suivant :



- a) Que représente ce diagramme ? quel est son intérêt principal ?
- b) Expliquer les éléments de base de ce diagramme.
- c) Quel est le **style architectural (Patron Architectural)** décrit par ce diagramme ?

2. Soit le diagramme UML suivant :



- Que représente ce diagramme? quel est l'intérêt principal d'un tel diagramme?
- Expliquer les éléments de base de ce diagramme, notamment les **cercles**, les **demi-cercles** et leur **jonction**?

### Exercice 2 (8 points):

Les systèmes d'exploitation tels que **Microsoft Windows**, **Mac OS** et **Linux** supportent des **interfaces graphiques**. On vous demande de proposer une modélisation d'un système permettant la **création de différents** composants d'une interface graphique (**boutons**, **barres de défilement** et **menus**) en fonction du **système d'exploitation** auquel ils sont destinés. Le modèle doit permettre de créer des composants **sans devoir connaître leurs classes concrètes**. Ainsi, le système doit être **indépendant** de la façon dont les **composants évoluent**.

- Quel est l'intérêt principal des  **patrons de conception (Design Pattern)** ?
- Quel est le  **patron de conception**  le mieux adapté pour répondre au problème ci-dessus? quel est son type ?
- En appliquant le  **patron de conception choisi**  dans la question précédente, proposer une modélisation du système demandé sous forme de  **diagrammes de classes** .
- Y-a-t-il un autre moyen, favorisant la réutilisation des composants et permettant de modéliser cette variation de systèmes d'exploitation, autre que les patrons de conception ?

**Bonne chance**