

# HEIG-VD — POO

## Laboratoire 3 (UML) – Compte-rendu

[redacted]  
20 octobre 2023

Ce document décrit nos choix et hypothèses de travail effectués pour réaliser le laboratoire n° 3 dans lequel nous avons dû créer un diagramme de classes UML permettant de représenter une agence de voyage.

### 1 Choix et hypothèses de travail

#### 1.1 Agence de Voyage

La récupération des voyages dans lequel une personne a participé se fait par le biais des employés. Comme un employé ne change pas d'agence, nous avons préféré cette logique. Ainsi, c'est plus simple au niveau des associations, même si l'on reconnaît que l'implémentation ne serait pas forcément la plus optimale.

#### 1.2 Client

Comme la donnée mentionne « Un client qui désire réserver un voyage se rend dans une agence. » ce qui, selon notre interprétation, sépare la notion de participant au voyage et client de l'agence de voyages ; par exemple, un client peut réserver un voyage pour un groupe tiers, sous la forme d'un cadeau.

Notre modèle sépare donc la notion de participants d'un voyage à la notion de client d'une agence sous deux associations pas nécessairement dépendantes.

Cela permet aussi à un employé d'une agence de réserver un voyage à titre privé dans son agence ou dans une autre.

##### 1.2.1 Personne

Par ailleurs, notre classe **Personne** n'est pas abstraite, car elle regroupe toutes les informations nécessaires pour identifier les clients et participants. Les champs sont néanmoins *protégés*, pour laisser une marge de manœuvre aux potentielles spécialisations de la classe concernant la manipulation des données.

Une de ces spécialisations est la classe **Employé** qui enregistre le salaire mensuel ainsi que la méthode de récupération des voyages organisés par ledit employé.

Le champ « téléphone » a été modélisé avec un *string*, afin de permettre l'utilisation des indicateurs téléphoniques, les clients et participants d'une agence pouvant provenir de divers pays.

#### 1.3 Voyage

Le numéro d'un voyage est l'identifiant de la réservation, modélisé sous forme de *string*, car nous estimons qu'un identifiant alphanumérique est probablement meilleur dans le cadre d'une gestion complexe comme pour celle d'une agence de voyage.

##### 1.3.1 Participants d'un voyage

La personne qui réserve un voyage ne fait pas forcément partie des participants. Nous ne l'avons pas mentionné dans le diagramme puisque c'est plutôt une contrainte d'implémentation.

#### 1.4 Hébergements

Chaque hébergement possède une adresse dans une ville précise. La ville peut donc recenser tous ses hébergements et permet d'en extraire la liste des **Hôtels** dont le nombre d'étoiles est égal à 3.

##### 1.4.1 Hôtels

Le nombre d'étoiles est, comme mentionné dans la donnée, de 1 à 5. Cela n'est pas mentionné dans le diagramme, car c'est une contrainte d'implémentation.

##### 1.4.2 Rabais

Nous avons imaginé le rabais comme étant utilisé lors du calcul du prix pour une personne donnée. On pourrait imaginer une fonction pour tout hébergement qui prend en paramètres une personne et détermine si les critères d'un des rabais offerts est applicable.

Cela permettrait, en outre, d'ajouter d'autres types de rabais applicables à d'autres tranches d'âge.

## 2 Diagramme

