

# ***Rapport UML***

***BERTHORELLY Lea***  
***GDIRA Ghizlane***  
***SY SAVANE Ahmadou Dia***

***Licence d'Informatique***  
***Ingénierie Logicielle***  
***UE Modélisation Objet et UML***

## Table of Contents

<b>Introduction.....</b>	<b>3</b>
<b>Diagramme de cas d'utilisation.....</b>	<b>4</b>
<b>Diagramme de classe .....</b>	<b>5</b>
<b>Diagramme d'états-transitions .....</b>	<b>6</b>
<b>Diagramme de séquences.....</b>	<b>8</b>
<b>Planning .....</b>	<b>10</b>

## **Introduction :**

Ce rapport contient les différents diagrammes avec leurs descriptions textuelles que nous avons fait pour à la réalisation de ce projet.

Il contient cinq parties :

- Première partie : le diagramme de cas d'utilisation.
- Deuxième partie : le diagramme de classe.
- Troisième partie : le diagramme d'états-transitions.
- Quatrième partie : le diagramme de séquences.
- Cinquième partie : le planning.

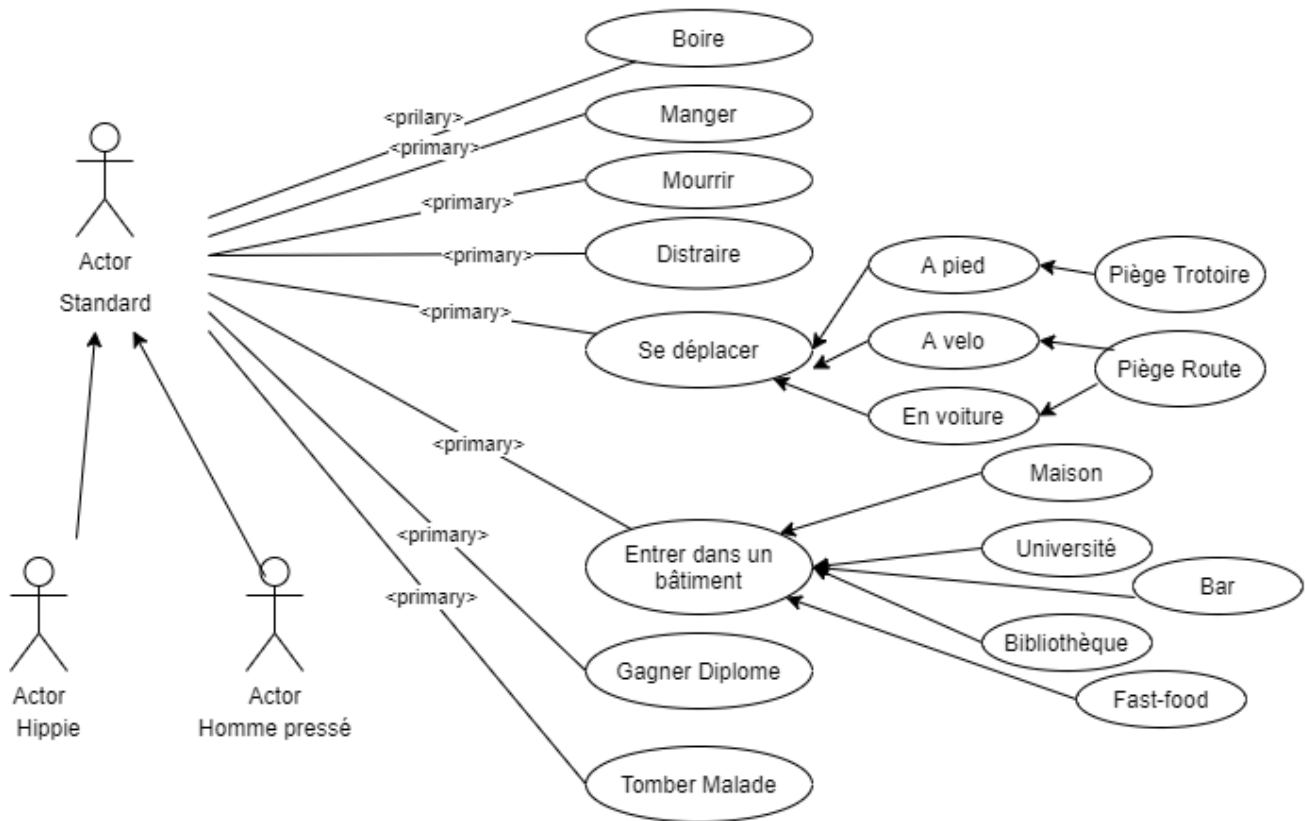
## I. Diagramme de cas d'utilisation :

### → Identification des acteurs :

Dans le jeu qu'on a créé nous avons un seul acteur qui est le personnage (standard, hippie et pressé), vu que le personnage est le seul qui interagit directement avec le système.

### → Les Cas d'utilisation :

- Boire
- Manger
- Mourir
- Distraindre
- Se déplacer
- Entrer dans un bâtiment
- Gagner diplôme
- Tomber malade



## II. Diagramme de classes

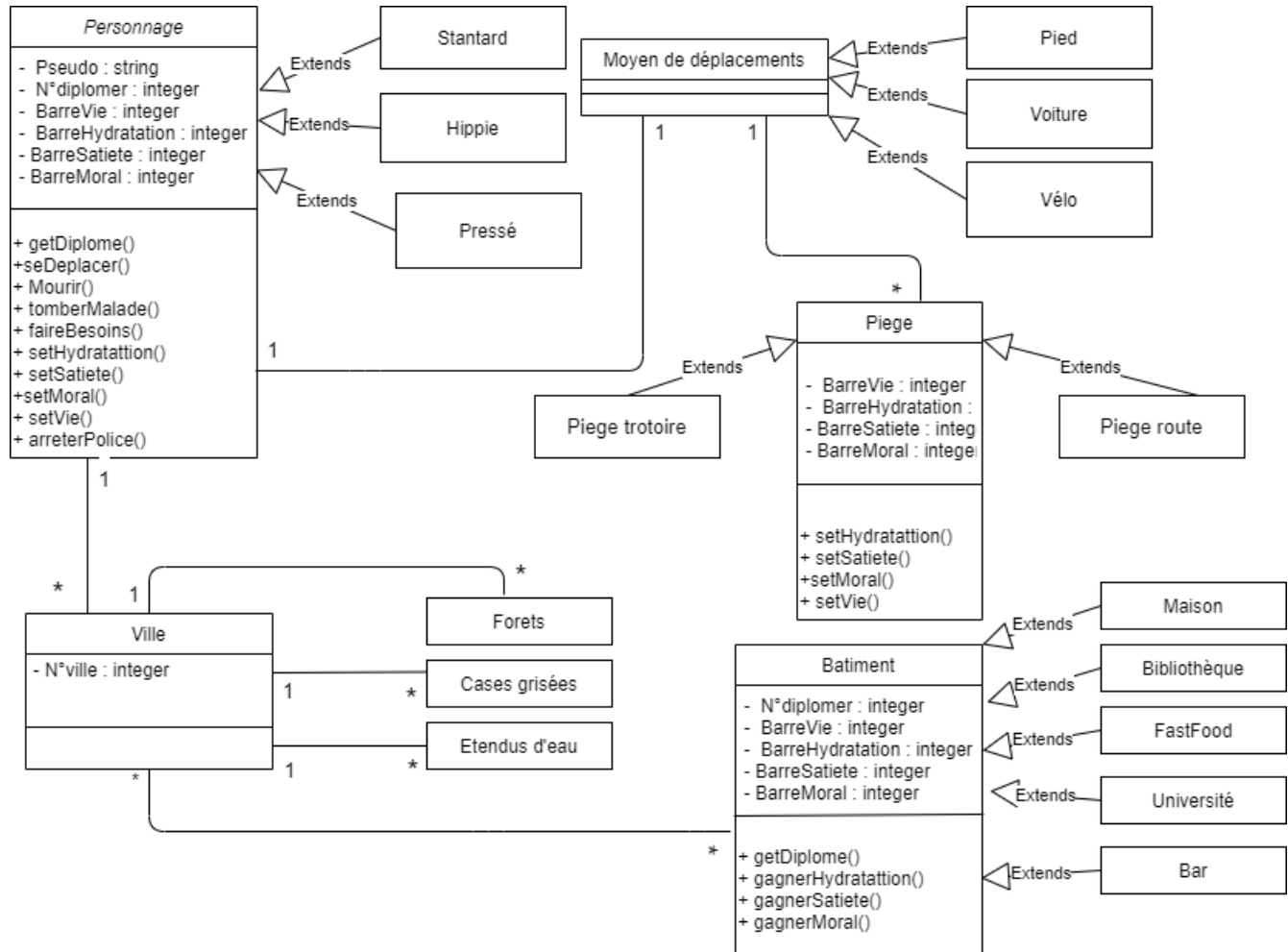
Le diagramme de classes suivant modélise la vie d'un personnage au sein d'une ville avec objectif d'obtenir le plus de diplômes possibles.

Il représente la structure des classes et les différentes relations entre elles. Il contient donc l'ensemble des informations qui sont gérées par le système.

Deux types de relations sont utilisés :

→ Extend : par exemple entre la classe Personnage et ses types.

→ Association : par exemple entre la classe Personnage et Ville.



On a commencé par la classe Personnage avec ces trois types :

→ Standard

→ Hippie

→ Pressé

Chaque action fait perdre Points variant pour chaque personnage.

Le personnage a la possibilité de se déplacer de 3 façons différentes : à pied, à vélo, ou en voiture, et il peut tomber sur 2 types de pièges : trottoir ou route.

Le personnage a la possibilité d'entrer dans une maison, bibliothèque, Fastfood, université et le bar.

### III. Diagramme d'états-transitions :

Pour un premier temps, lorsque l'on lance la simulation le personnage se trouve dans Maison d'où état initiale de notre simulation. Une fois dans la maison le personnage peut se ressourcer (nourrir, boire, mangé), il peut se déplacer, peut tomber malade, peut mourir.

Lorsqu'il tombe malade il perd 10 points de sa barre de vie.

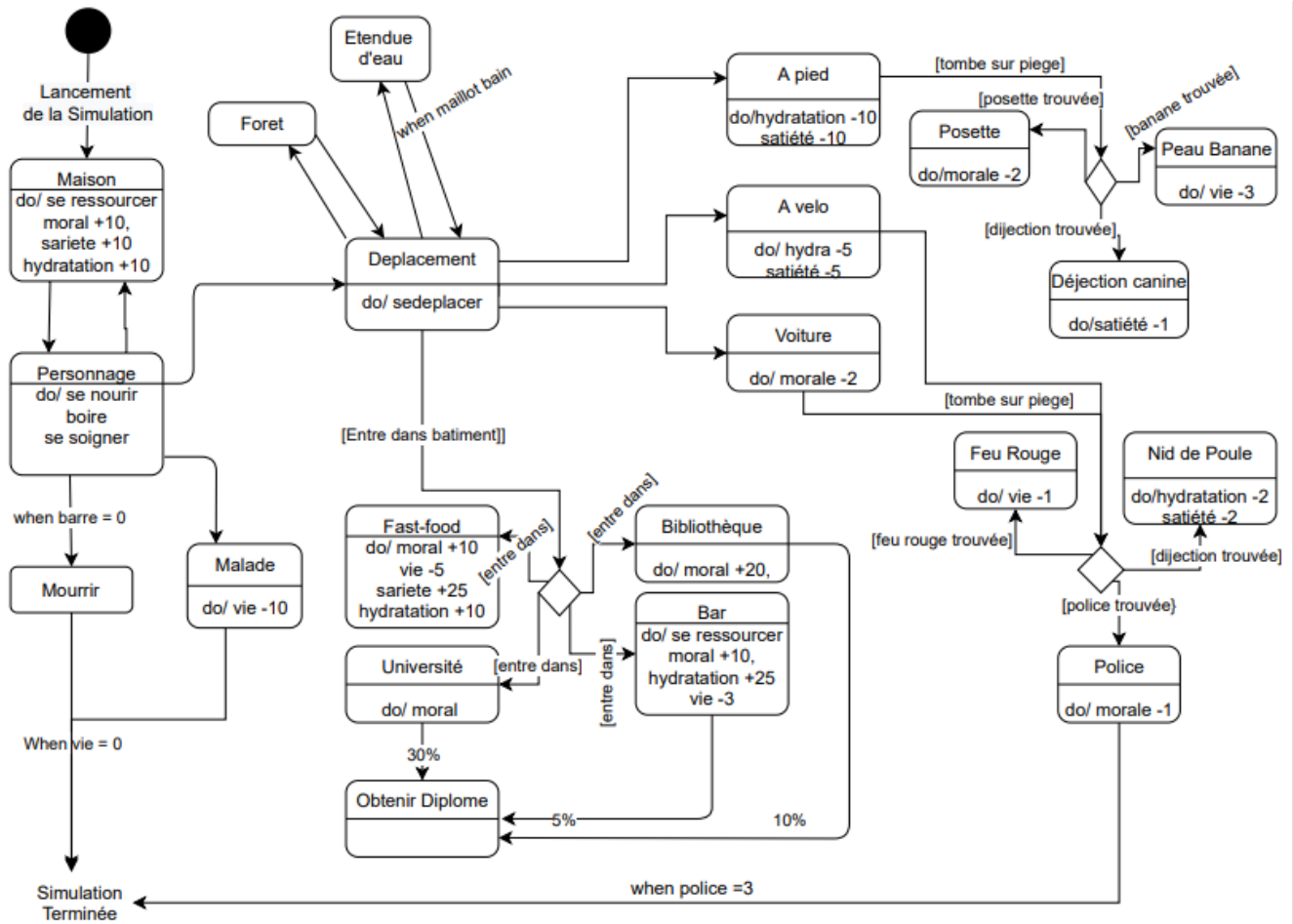
Peut Mourir lorsque toutes ces barres tombent à 0, lorsque sa barre de vie est à 0 la simulation est terminée d'où état finale.

Le personnage peut se déplacer dans les différents bâtiments comme il est décrit dans le diagramme, Fast-food, Université, Bibliothèque, Bar, Forêts, Etendue d'eau s'il a un maillot de bain et il a la chance d'obtenir le diplôme en fonction du bâtiment.

Le personnage peut se déplacer à Pied, à Vélo et en Voiture, il perd des points selon le mode de déplacement qu'il a choisis.

Pour chaque mode de déplacement nous avons des pièges, soit des pièges trottoir (Poussette, Peau de banane, Déjection canine) s'il est à pied, soit des pièges route (Feu rouge, Police, Nid de poule) s'il choisit de se déplacer à vélo ou à voiture.

Il peut perdre des points quand il tombe sur un piège. Lorsqu'il tombe sur la police la simulation est terminée.



#### IV. Diagramme de séquences :

Chaque action fait appel à la fonction perdrePoints(n), avec n variant pour chaque personnage.

L'homme standard perd 1 point sur chaque barre, le hippie perd 0,5 points sauf pour les déplacements qui lui en font perdre 2. De plus, son moral ne baisse jamais.

Enfin, l'homme pressé perd 1 point par barre sauf en cas de déplacement où ses barres ne perdent aucun point à l'exception de la barre de moral qui en perd 2.

Dans le diagramme de séquence de déplacements, le personnage a la possibilité de se déplacer de 3 façons différentes : à pied, à vélo, ou en voiture.

Si le personnage choisit de marcher, chaque déplacement fait appel à deux fonctions : perdreHydratation(n) et perdreSatiété(n).

Si le déplacement choisi est la voiture, 3 fonctions sont appelées : risquerMort(n), perdreMoral(n), et risquerArrestation(n). Seul le hippie ne peut pas utiliser ce moyen de transport.

Enfin pour le vélo, 3 fonctions sont également appelées : perdreHydratation(n), perdreSatiété(n) et risquerMort(0,5).

Selon le type de personnage, n pourra prendre des valeurs différentes.

L'homme pressé ne peut pas perdre de point de santé, d'hydratation et de satiété, et ce, peu importe son type de déplacement. En revanche, il perd 2 points de moral à chaque déplacement.

Quant au hippie, il perd 2 fois plus de points que l'homme standard.

Dans le diagramme de séquence lié aux bâtiments, entrer dans la maison fera appel aux fonctions gagnerMoral(10), gagnerSatiété(10), et gagnerHydratation(10). Elles font chacune gagner 10 points aux barres concernées et ce, peu importe le personnage.

Entrer dans la bibliothèque fait appel à la fonction gagnerMoral(20) ainsi qu'à la fonction trouverLivreGL(5). Cette fonction permet de savoir si le personnage a trouvé le livre sur le génie logiciel et si celui-ci est trouvé, la fonction obtenirDiplôme(10) est appelée pour attribuer 10 % de chance d'obtenir un diplôme.

Le fast-food fait appel à la fonction gagnerSatiété(25).

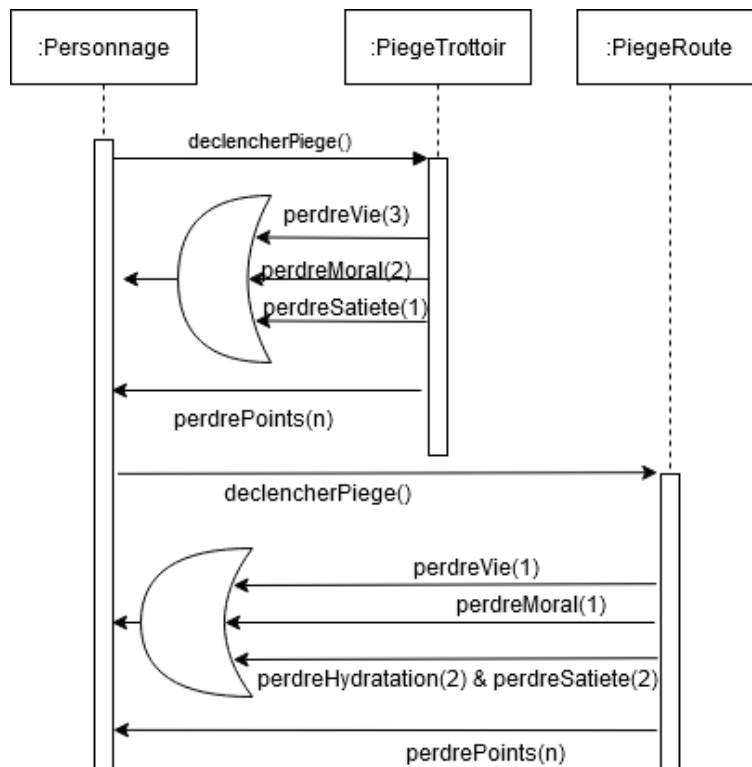
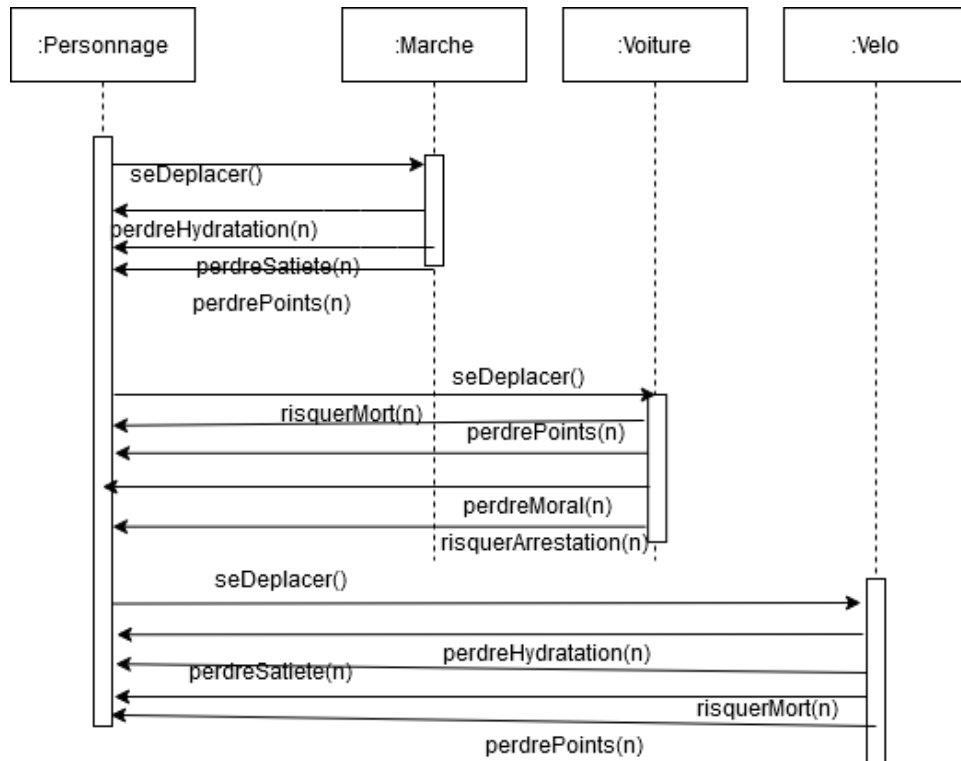
L'université fait quant à elle appel à la fonction obtenirDiplôme(30). Elle permet d'attribuer 30 % de chance d'obtenir un diplôme.

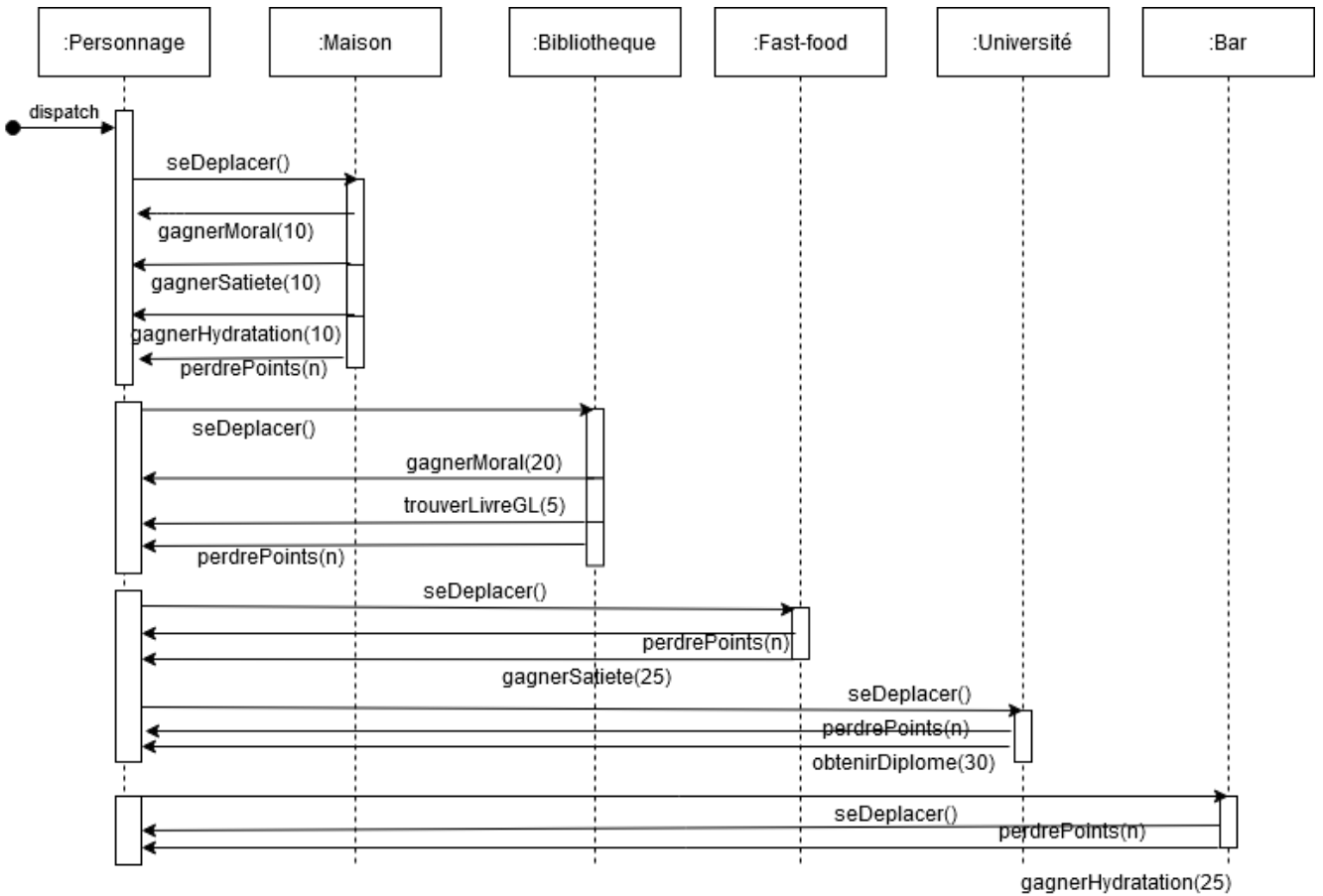
Enfin, entrer dans le bar fait appel à la fonction gagnerHydratation(25).

Enfin, dans le diagramme de séquence des pièges, 2 types de pièges sont possibles. Si le personnage tombe sur un piège trottoir, un « tirage au sort » est effectué afin de savoir quelle fonction est appelée. Les 3 fonctions sont : perdreVie(3), perdreMoral(2), et perdreSatiété(1).

Au contraire, si le personnage tombe sur un piège route, les 3 événements sont : perdreVie(1), perdreMoral(2) et perdreHydratation(2) + perdreSatiété(2).







## V. Planning :

Pour ce qui est du planning mis à jour, il sera joint comme tous les diagrammes et les fichiers qui seront déposés sur la plateforme et GITHUB.

