

# PLD AGILE

Rapport Première Itération

H4103

# SOMMAIRE

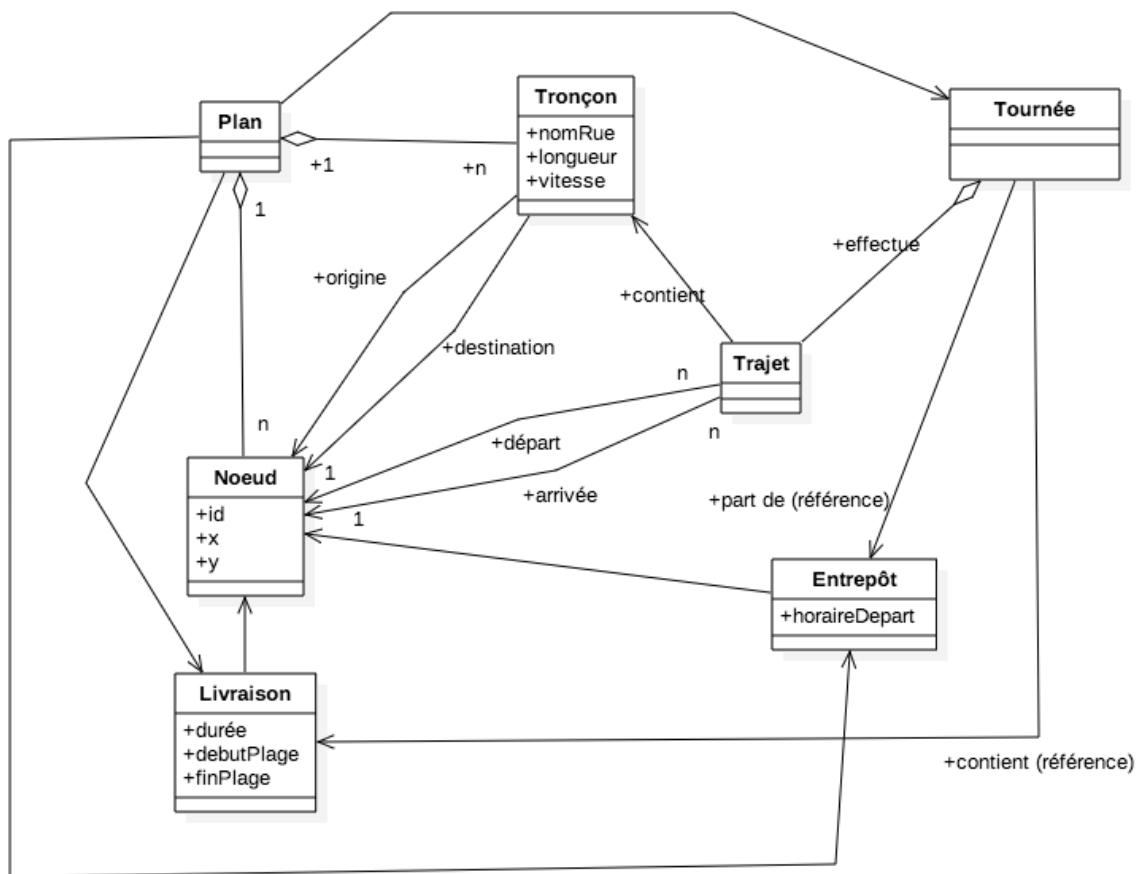
<b>Introduction</b>	<b>3</b>
<b>Modèle de domaine</b>	<b>3</b>
<b>Glossaire</b>	<b>4</b>
<b>Diagramme de cas d'utilisation</b>	<b>5</b>
<b>Description textuelle des cas d'utilisation</b>	<b>6</b>
<b>Diagramme d'États-Transition</b>	<b>7</b>
<b>Diagramme de package</b>	<b>8</b>
<b>Diagramme de classes</b>	<b>9</b>
<b>Diagramme de séquence de calcul d'une tournée</b>	<b>10</b>
<b>Planning effectif de la première itération</b>	<b>11</b>
<b>Annexes</b>	<b>12</b>

# Introduction

## Equipe :

- **Nicolas Chazeau** : Scrum Manager - Developer
- **Jacques Folleas** : Product Owner - Developer
- **Quentin Vecchio** : Responsable qualité - Developer
- **Ludovic Cros** : Developer
- **Pierre Bayle** : Developer
- **Pierre Jaglin** : Developer
- **Nicolas Gripont** : Developer

## Modèle de domaine



# Glossaire

**Plan** : Carte de l'ensemble des noeuds et tronçons.

**Noeud** : Cela correspond à un point sur le plan (coordonnées X, Y), il a un identifiant unique.

**Adresse** : Cela correspond à un noeud.

**Tronçon** : Une arête, segment de route. Qui a une origine (noeud de départ), une destination (noeud d'arrivée), cela détermine donc un sens. Contient également une longueur, vitesse et nom de rue.

**Trajet** : Suite de tronçons qui permet d'aller d'une livraison de départ à une livraison d'arrivée.

**Entrepôt/dépôt** : Noeud de départ de la tournée. Contient une heure de départ.

**Tournée** : Liste triée des livraisons à effectuer. Une suite de trajets est ensuite calculée permettant d'obtenir un ordre de livraison.

**Livraison** : Noeud à livrer, contient une durée de livraison, et éventuellement une plage horaire.

**Coût** (d'une tournée) : temps total que va prendre le parcours total d'une tournée ainsi que les livraisons à effectuer.

**Noeud Départ** : C'est un noeud correspondant au point de départ d'un tronçon.

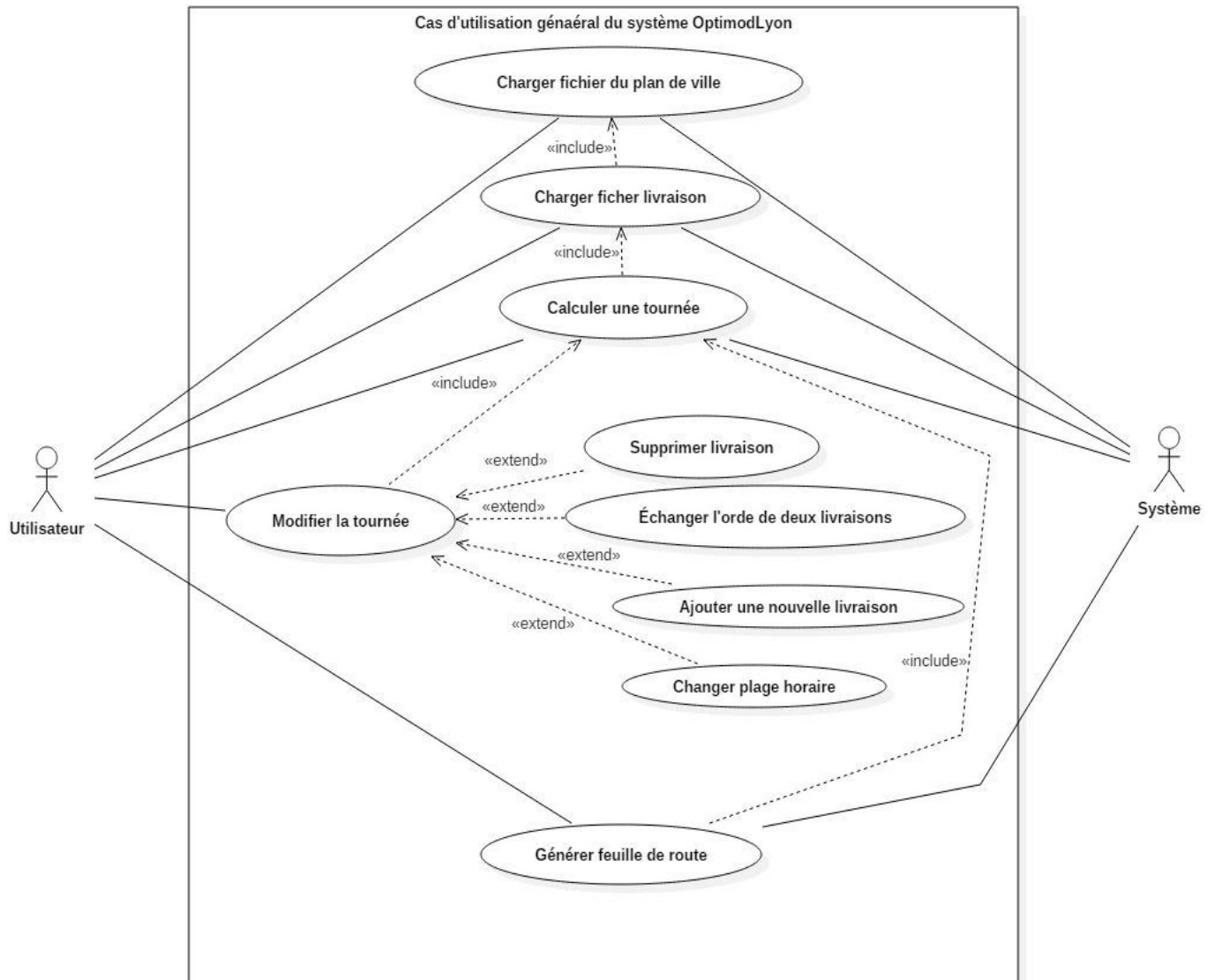
**Noeud Arrivée** : C'est un noeud correspondant au point d'arrivée d'un tronçon.

**Feuille de route** : Correspond à un fichier donnant la liste des livraisons dans l'ordre à effectuer.

**Origine** : C'est un noeud de départ d'un tronçon.

**Destination** : C'est un noeud d'arrivée d'un tronçon.

# Diagramme de cas d'utilisation



# Description textuelle des cas d'utilisation

## Cas n°1 : Charger fichier plan de la ville

**Précondition :** L'utilisateur a lancé l'application

**Scénario Principal :**

1. Le système affiche une fenêtre contenant l'explorateur de fichier.  
1Bis. L'utilisateur dépose un fichier XML via un Glisser-Déposer.
2. Choisit un fichier XML depuis l'explorateur de fichier.
3. Le système affiche le plan de la ville chargé dans la carte.

**Scénario Alternatif :**

- 2a. Le fichier choisi n'est pas un fichier XML;  
Le système indique à l'utilisateur que le fichier n'est pas au format XML et retourne à la page d'accueil/démarrage.
- 2b et 1Bis. Le fichier choisi contient des données erronées et/ou "mal écrites".  
Le système indique que le fichier ne peut être ajouté en raison d'un problème de lecture et retourne à la page d'accueil/démarrage.

## Cas n°2 : Charger fichier livraison

**Précondition :** un plan de la ville a été chargé

**Scénario Principal :**

1. Le système affiche une fenêtre contenant l'explorateur de fichier.  
1Bis. L'utilisateur dépose un fichier XML via un Glisser-Déposer.
2. Choisit un fichier XML depuis l'explorateur de fichier.
3. Le système affiche la demande de livraison chargée dans la carte contenant le plan de la ville, et de manière textuelle dans le tableau.

**Scénario Alternatif :**

- 2a. Le fichier choisi n'est pas au format XML;  
Le système indique à l'utilisateur que le fichier n'est pas au format XML et retourne à la page affichant le plan de la ville.
- 2b et 1Bis. Le fichier choisi contient des données erronées et/ou "mal écrite".  
Le système indique que le fichier ne peut être ajouté en raison d'un problème de lecture et retourne à la page affichant le plan de la ville.

## Cas n°3 : Calculer une tournée

**Précondition :** L'utilisateur a chargé un plan de la ville et un plan de livraison

### Scénario Principal :

1. L'utilisateur appuie sur le bouton dédié à la procédure de calcul d'une tournée de livraison.
2. Le système calcul la tournée de livraison via une procédure de calcul puis affiche l'itinéraire de livraison dans le plan et la liste des livraisons (adresse, plage horaire début, plage horaire fin, durée) dans le tableau se trouvant à gauche du plan.

### Scénario Alternatif :

2a Il n'existe pas de tournée respectant les contraintes liées aux plages horaires.

Le système ne prend pas en compte le calcul de la tournée et le signale à l'utilisateur via un message dans l'interface. Le système retourne à la page principale avec le plan et la liste des livraisons.

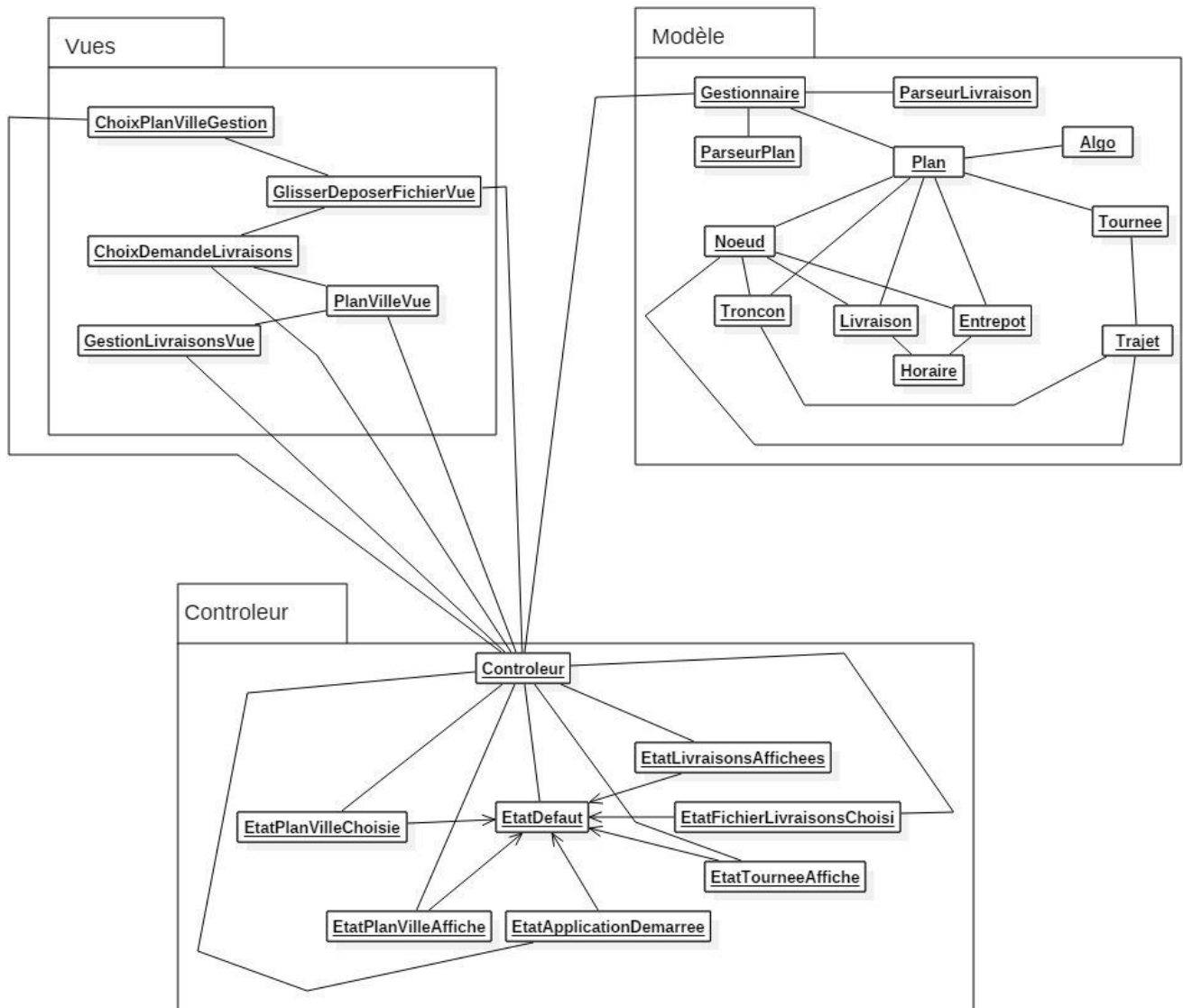
1-2a L'utilisateur indique au système qu'il souhaite annuler le calcul de la tournée.

Le système annule le calcul de tournée en cours

## Diagramme d'États-Transition

*Voir Annexe 1 - Diagramme d'États-Transition.*

## Diagramme de package



## Diagramme de classes

Voir Annexe 2 - Diagramme de classes du contrôleur.

Voir Annexe 3 - Diagramme de classes du domaine.

Voir Annexe 4 - Diagramme de classes de vue.

## Diagramme de séquence de calcul d'une tournée

Annexe 5 - Diagramme de séquence de calcul d'une tournée.

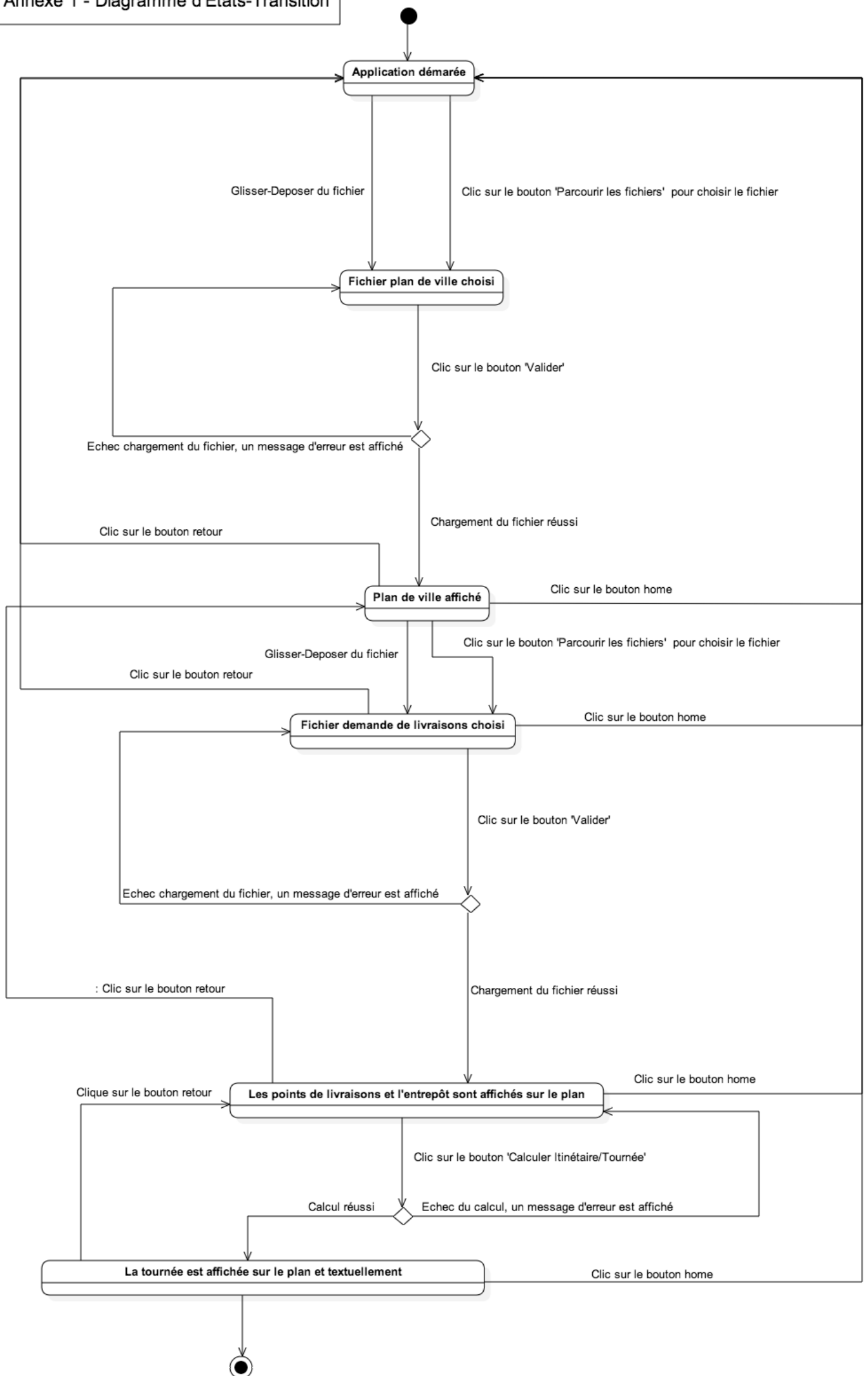


# Planning effectif de la première itération

Membres	Jacques F.	Ludovic C.	Nicolas C.	Nicolas G.	Pierre B.	Pierre J.	Quentin V.	TOTAL h/P
Tâches								
Diagramme modèle du domaine		1	4		4			9
Glossaire			1		1			2
Diagramme cas d'utilisation (CU)						4	2	6
Description textuelle CU						4		4
Diagramme États-transitions		2		2,5				4,5
Diagramme de classe modèle		2	1		4		1	8
Diagramme de classe controleur		4			4			8
Diagramme de classe vue				1			1	2
Diagramme de packages		1					1	2
Recherche Algorithme	8							8
Diagramme de séquence calcul d'une tournée	2							2
Mockups		3		3				6
Implémentation des pages (et prise en main JavaFX)				8			6	14
Implémentation algorithme de calcul	8		4	2				14
Implémentation parseurs			3			5		8
Implémentation modèle domaine			2	0,5			3	5,5
Implémentation contrôleur		4		1	4		1	10
Tests JUnit (unitaires et fonctionnels)						4	3	7
Planning + Taiga			2					2
<b>TOTAL h/P</b>	<b>18</b>	<b>17</b>	<b>17</b>	<b>18</b>	<b>17</b>	<b>17</b>	<b>18</b>	<b>122</b>

# Annexes

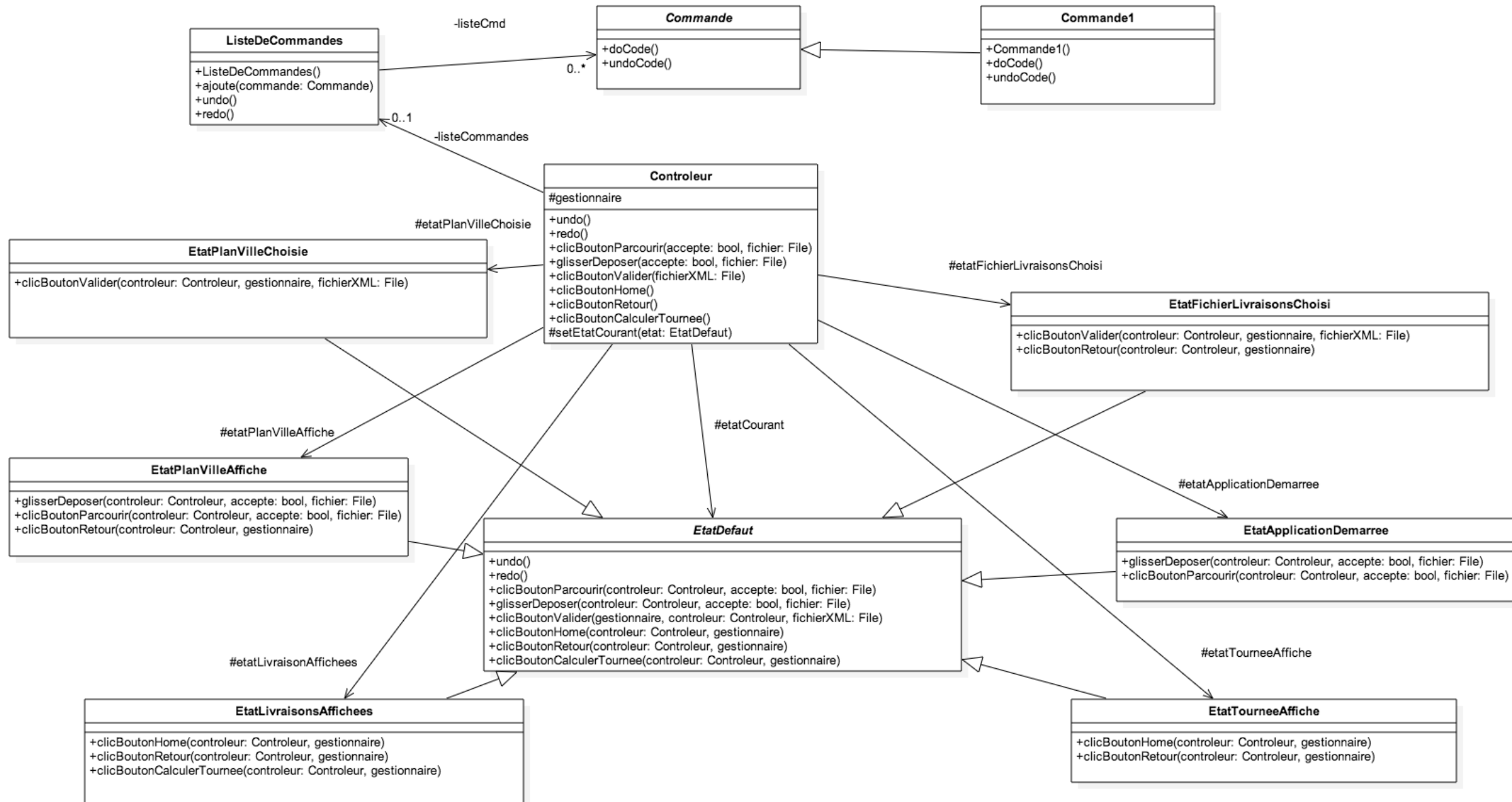
# Annexe 1 - Diagramme d'États-Transition



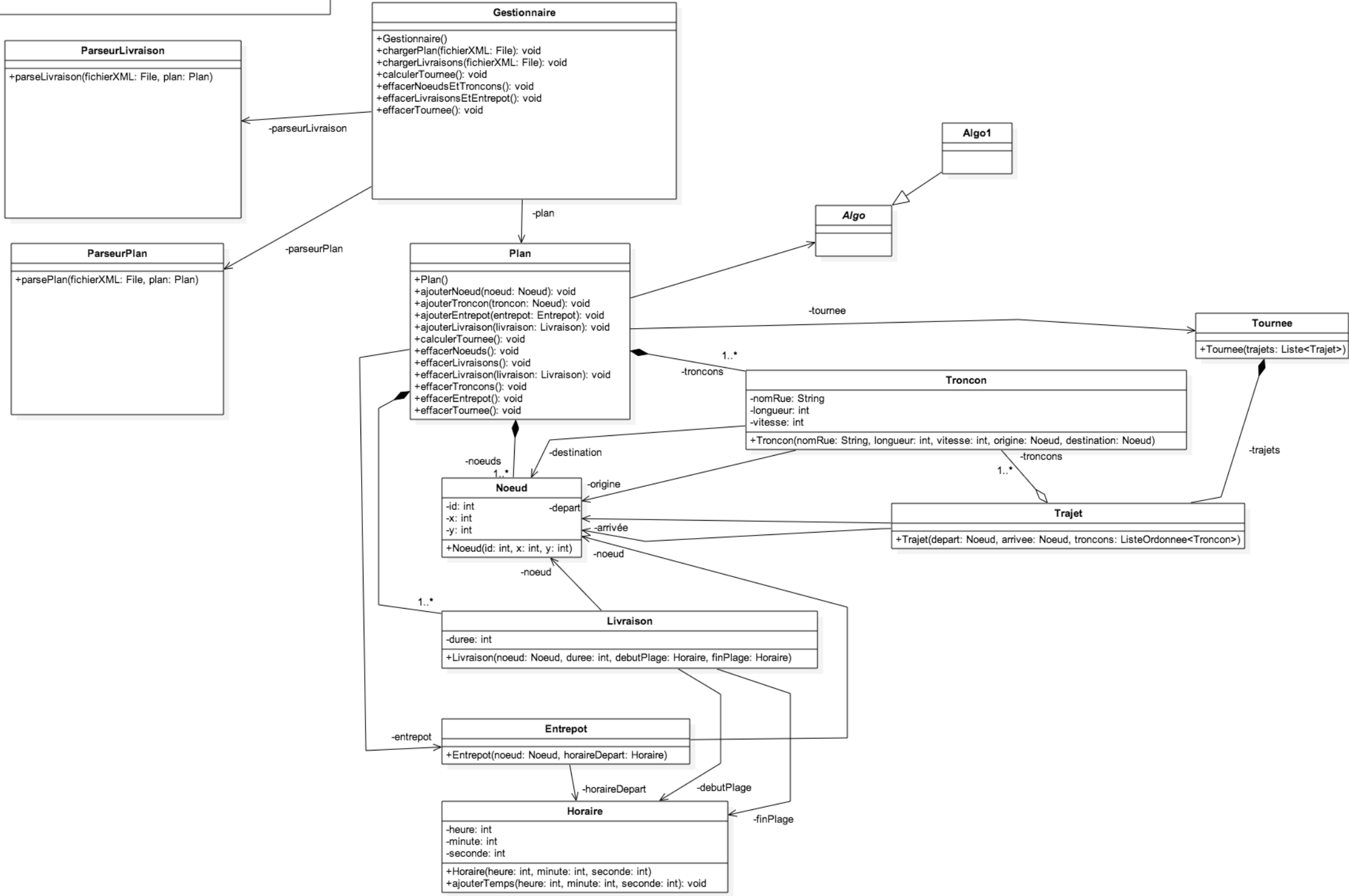
## Annexe 2 - Diagramme de classes du contrôleur

le controleur possède une instance de chaque etat ainsi qu'un etat courant. On modifie son etat courant dans les methodes etats en utilisant le controleur en paramètre et la méthode setEtatCourant

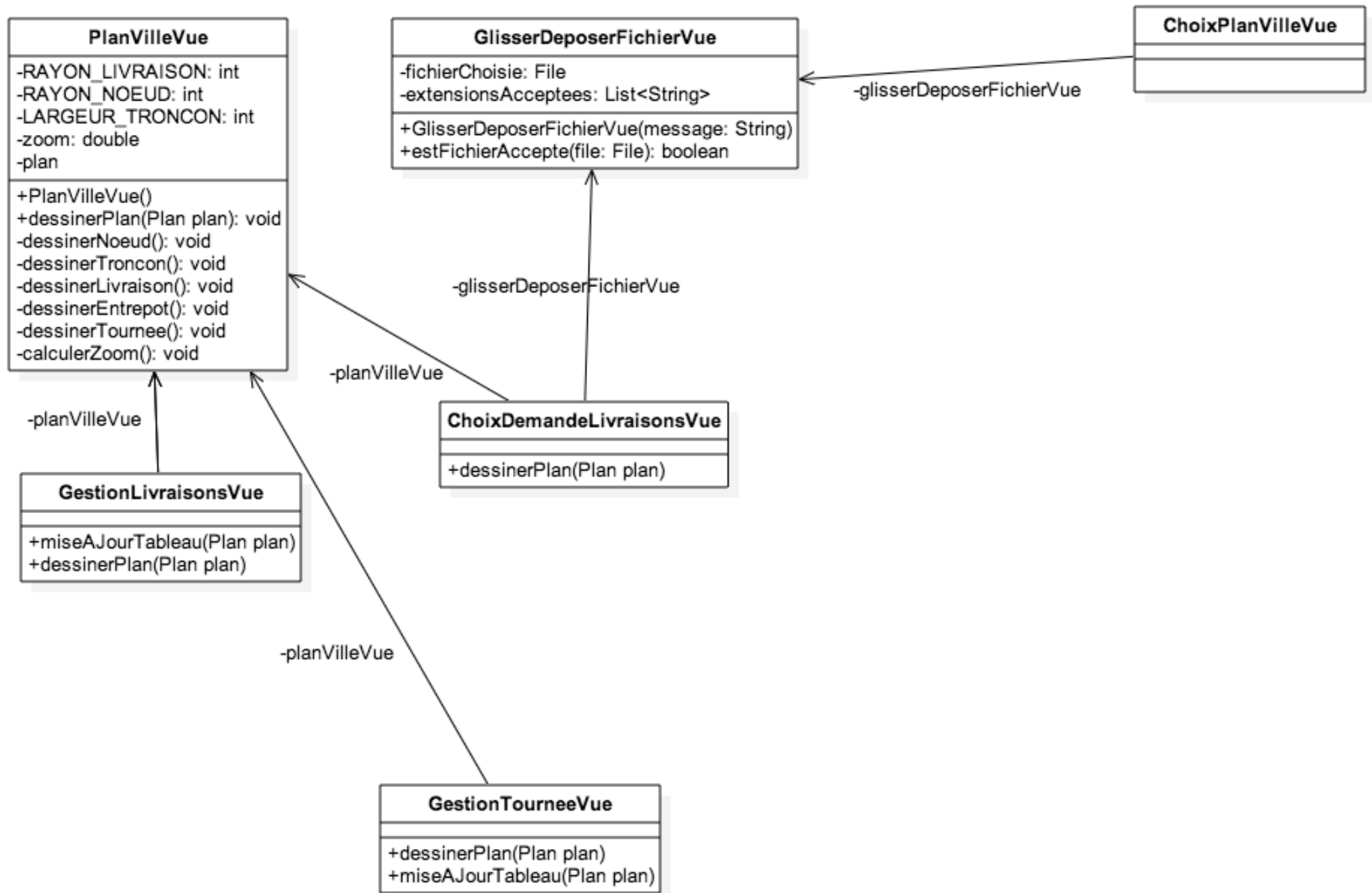
TODO : voir les commandes dans l'itération 2 avec l'ajout des modifications



Annexe 3 - Diagramme de classes du domaine.



## Annexe 4 - Diagramme de classes de vue



## Annexe 5 - Diagramme de séquence de calcul tournée

