Analyse

Table des matières

[I - Définition du sujet 2](#_Toc452048010)

[II - Dictionnaire des termes utilisés 2](#_Toc452048011)

[III - Descriptif de l’équipe 2](#_Toc452048012)

[1) Les acteurs 2](#_Toc452048013)

[Martin Bolot (Chef de projet) 2](#_Toc452048014)

[Yoann Merle 2](#_Toc452048015)

[Karim Oubah 2](#_Toc452048016)

[Arthur Zicaro 2](#_Toc452048017)

[2) Répartition des lots 3](#_Toc452048018)

[IV - Outils choisis 3](#_Toc452048019)

[1) NetBeans 3](#_Toc452048020)

[2) Oracle 3](#_Toc452048021)

[3) ArgoUML 3](#_Toc452048022)

[4) PowerAMC 3](#_Toc452048023)

[5) AnalyseSI 3](#_Toc452048024)

[6) Git / GitHub 3](#_Toc452048025)

[V - Diagrammes 4](#_Toc452048026)

[1) Diagramme UML de classes 4](#_Toc452048027)

[2) Diagramme de Gantt 4](#_Toc452048028)

[3) Diagramme de séquence 5](#_Toc452048029)

[4) Diagramme MCD (modèle conceptuel de données) 5](#_Toc452048030)

[VI - Planning et rôle des intervenants 6](#_Toc452048031)

# I - Définition du sujet

Ce cours application de synthèse va nous permettre de concevoir une application dans son ensemble en passant par toutes les phases de conception et de développement nécessaires.

Le sujet est le suivant : Nous devons assurer la conception et la mise en œuvre d’un logiciel de gestion des fédérations qui ont en charge l’organisation et la gestion des différentes compétitions de football en Europe.

# II - Dictionnaire des termes utilisés

POO : Programmation Orientée Objet

UML : Unified Modeling Langage, langage de modélistation des données

MCD : Modèle Conceptuel de Données

IDE : Intelligent Developpement Environement, environnement de développement intelligent

SGBD : Système de Gestion de Bases de Données

# III - Descriptif de l’équipe

## 1) Les acteurs

Par ordre alphabétique :

### Martin Bolot (Chef de projet)

Disposant d’une expérience de trois ans dans le domaine du développement Web, il a été choisi comme chef de projet car il a déjà pu observer dans un cadre professionnel le cycle de vie d’une application. Il s’occupera de la coordination de l’équipe, de la rédaction de certains documents fonctionnels mais aussi du développement d’une partie de l’application.

### Yoann Merle

Très familier avec la POO et avec le langage UML, il s’occupera de la définition du diagramme de classes et de son implémentation concrète dans l’application, mais aussi de veiller à la cohérence de la gestion des classes par rapport au MCD.

### Karim Oubah

Karim vient d’une filière GEA et possède de solides connaissances en gestion, et sera en charge d’une partie de la gestion des plannings. Son projet tuteuré est basé sur le Java, il sera donc le référent technique pour toutes les questions ayant trait au développement de l’application Java et à la couche d’accès aux données.

### Arthur Zicaro

Arthur est issu du milieu scientifique et plus précisément d’une filière mathématique, son domaine de compétence porte vers l’algorithme. Il aura donc pour rôle, entre autres, de définir en amont les couches logiques qui seront utilisés dans les différentes parties de l’application.

## 2) Répartition des lots

Deux sous-groupes ont été formés pour la répartition des tâches :

* Martin et Yoann s’occuperont des championnats nationaux
* Arthur et Karim s’occuperont des championnats internationaux

# IV - Outils choisis

## 1) NetBeans

Nous avons choisi NetBeans comme environnement de développement pour notre application Java. En effet, il s’agit d’un IDE gratuit et performant, installé sur toutes les machines sur lesquelles nous allons trailler.

## 2) Oracle

Nous allons utiliser Oracle comme SGBD pour la gestion des données persistantes de l’application. Ce logiciel sera utilisé pour deux raison : premièrement l’équipe en a une bonne connaissance car il a été étudié en cours via deux modules, et deuxièmement car il s’intègre facilement dans un environnements Java.

## 3) ArgoUML

ArgoUML est un outil gratuit qui nous servira à effectuer une partie de nos diagrammes UML.

## 4) PowerAMC

PowerAMC est un outil payant mais disponible sur les machines de l’IUT qui nous permettra de réaliser les diagrammes UML les plus complexes que nous aurons à produire (notamment le diagramme de classes), plus facilement faisable que sous ArgoUML.

## 5) AnalyseSI

Sera utilisé pour concevoir le modèle entité-association (MCD) de la base de donnée de l’application.

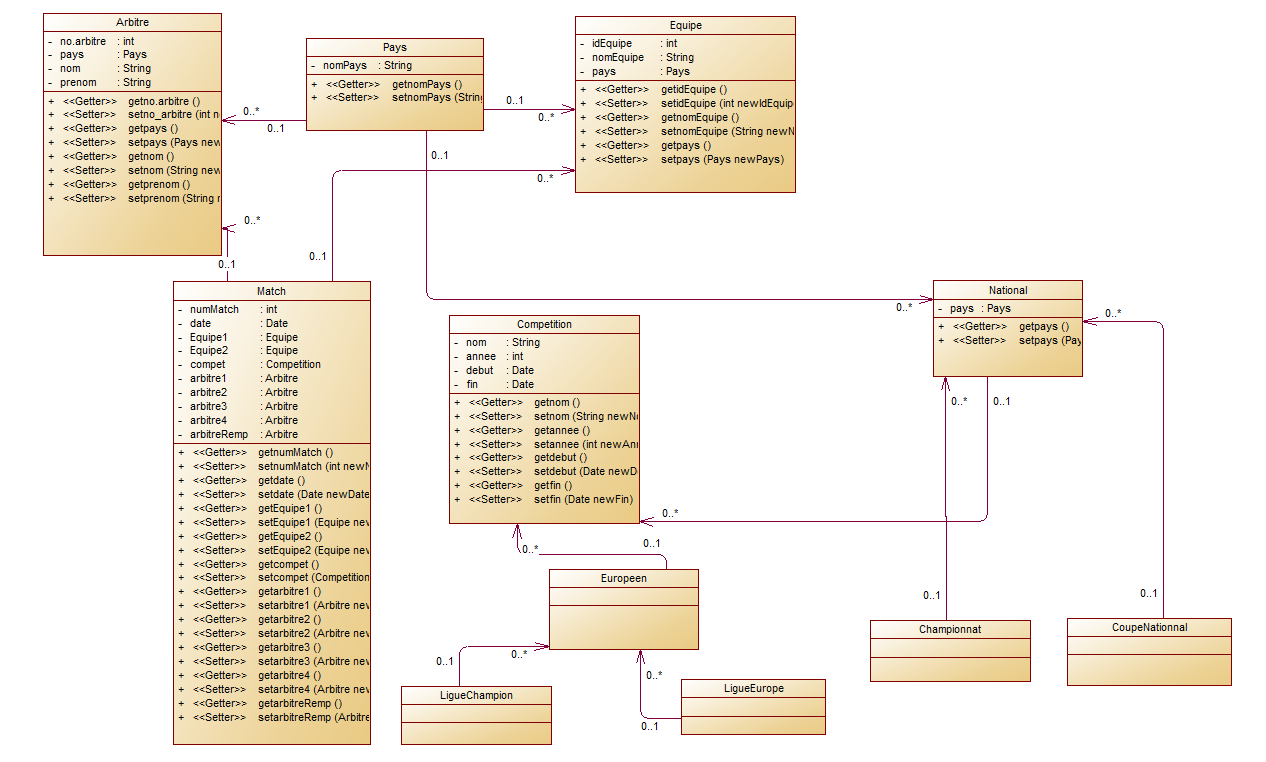
## 6) Git / GitHub

Nous utiliserons Git comme logiciel de gestion de version durant le développement de ce programme Java, ainsi que la plateforme GitHub afin d’héberger notre projet en open-source. Les outils utilisés pour Git seront la ligne de commande et l’interface graphique. Seules les sources du programme seront versionnées (dossier « src/ ») afin que chaque acteur du projet puisse participer en créant son propre projet NetBeans en local.

# V - Diagrammes

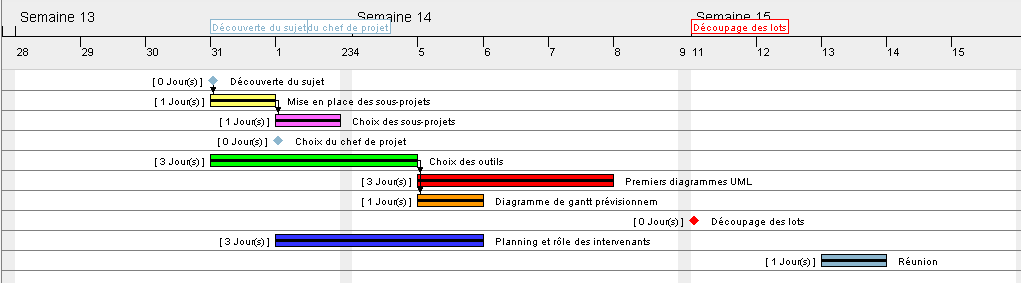
## 1) Diagramme UML de classes

Le diagramme de classe définit les interactions et relations entre les différentes classes.



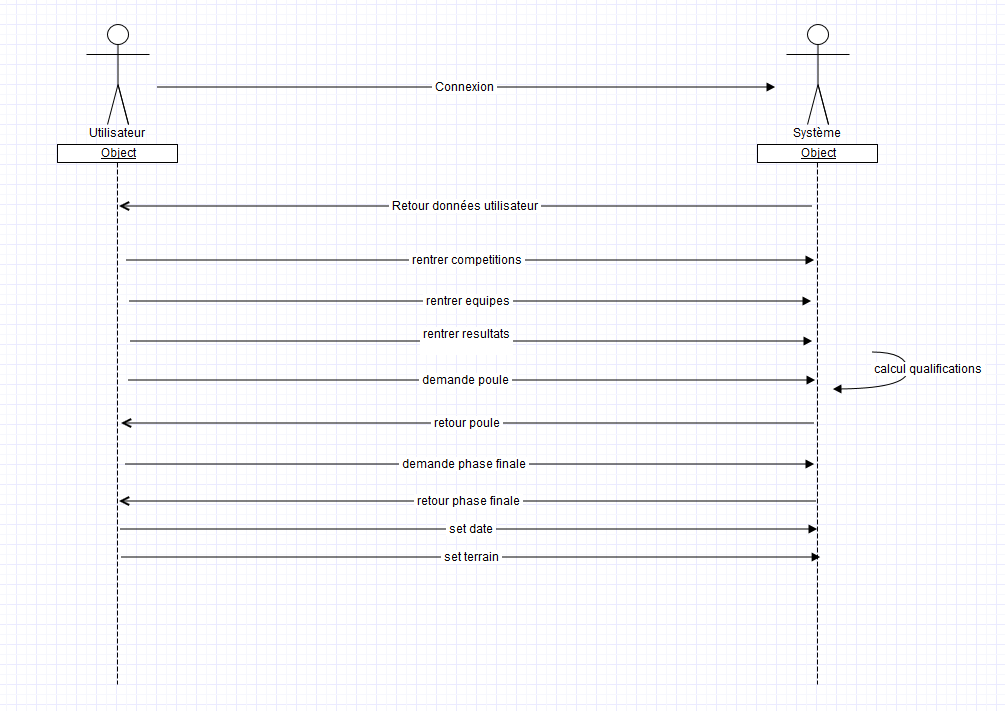
## 2) Diagramme de Gantt

Le diagramme de Gantt permet de représenter toutes les tâches définies par l’équipe et leur déroulement prévisionnel dans le temps (date de début et date de fin).



## 3) Diagramme de séquence

Le diagramme de séquence permet d’illustrer les acteurs de l’application et leurs interactions.



## 4) Diagramme MCD (modèle conceptuel de données)

Le diagramme MCD nous permet de définir la structure de notre base de données ainsi que les relations entre les différentes tables qui la composeront et leurs attributs.

# VI - Planning et rôle des intervenants

* Découvert du projet : 31/03
* Mise en place des sous-projets : 31/03
* Choix des sous-projets : 01/04
* Choix du chef de projet : 01/04
* Choix des outils : 31/03 - 04/04
* Premiers diagrammes UML : 05/04 – 07/04
* Diagrammes de Gantt : 05/04
* Découpage des lots : 11/04
* Planning et rôle des intervenants : 01/04 – 05/04
* Réunion : 13/04