

Projet LU2IN002 - 2020-2021

Numéro du groupe de TD/TME : 9

<i>Nom</i> : BA	<i>Nom</i> : SPAAK	<i>Nom</i> : -
<i>Prénom</i> : Danny	<i>Prénom</i> : Nelson	<i>Prénom</i> : -
<i>N° étudiant</i> : 3805136	<i>N° étudiant</i> : 28600545	<i>N° étudiant</i> : -

Thème choisi (en 2 lignes max.)

Bibliothèque de gestion d'un scoreboard ainsi que la génération de commentaires d'un match de basketball.

Description des classes et de leur rôle dans le programme (2 lignes max par classe)

TimeOutOfBoundsException : Exception personnalisée générée lorsque l'on dépasse le temps maximal affichable.

Time : Temps restant du match en cours.

Team : Rassemble les joueurs remplaçants et titulaires avec leurs différents postes.

SmallForward : Poste de joueur, Ailier en français.

Center : Poste de joueur, Pivot en français.

ShootingGuard : Poste de joueur, Arrière en français.

PowerForward : Poste de joueur, Ailier fort en français.

PointGuard : Poste de joueur, Meneur en français.

Scoreboard : Affiche les informations essentielles durant un match.

Player : Représente un joueur de basket-ball. Classe abstraite car un joueur de basket a besoin d'avoir un poste assigné en jeu.

PeriodOutOfBoundsException : Exception générée lorsqu'on dépasse le nombre maximal de quarts de temps.

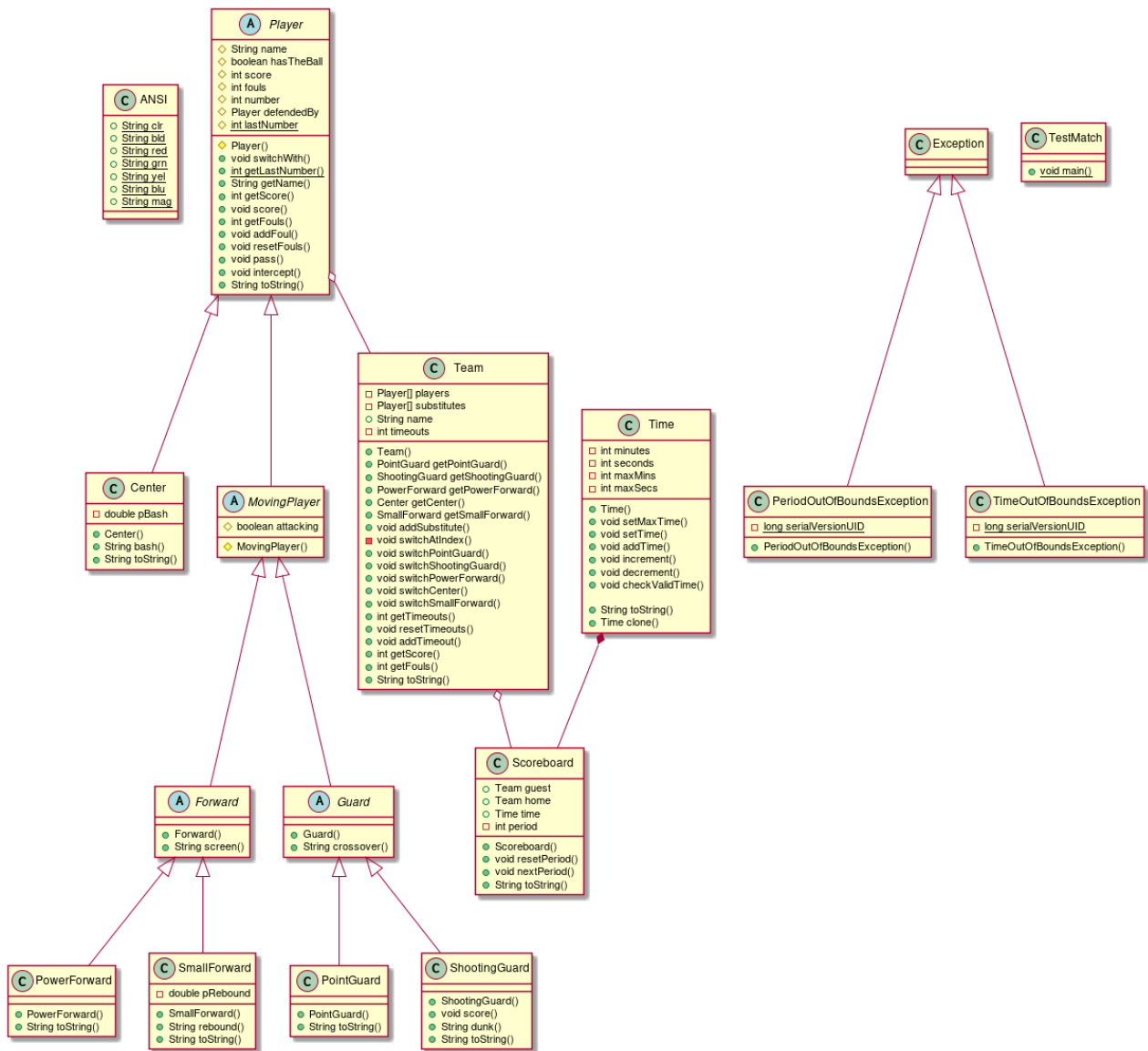
MovingPlayer : Représente un joueur capable d'attaquer et de défendre (tout sauf Center).

Guard, Forward : Classes abstraites qui aident à la personnalisation du poste des joueurs.

ANSI : Fournit les codes couleurs ANSI pour le terminal.

TestMatch : Exemple d'un match de deux équipes : notre client le commentateur souhaite utiliser notre programme pour commenter le match, mais aussi afficher son avancement avec le score.

Schéma UML des classes vision fournisseur (dessin “à la main” scanné ou photo acceptés)



<i>Checklist des contraintes prises en compte:</i>	<i>Nom(s) des classe(s) correspondante(s)</i>
Classe contenant un tableau ou une ArrayList	Team
Classe avec membres et méthodes statiques	TestMatch ANSI
Classe abstraite et méthode abstraite	Forward Guard MovingPlayer Player
Interface	-
Classe avec un constructeur par copie ou clone()	Time
Définition de classe étendant Exception	PeriodOutOfBoundsException TimeOutOfBoundsException
Gestion des exceptions	TestMatch
Utilisation du pattern singleton	-

<i>Présentation de votre projet (max. 2 pages) : texte libre expliquant en quoi consiste votre projet.</i>
<p>Notre supposé client souhaite commenter un match de basket-ball sans se fatiguer. Nous lui offrons la possibilité de commenter les actions les plus décisives du match tout en permettant au spectateur de voir son avancement, les différents joueurs, et surtout le score, sur un scoreboard affiché en couleurs.</p> <p>Nous fournissons aussi une logique basique telle que la possibilité pour un joueur de défendre sur un autre ou d'être en état d'attaque.</p>