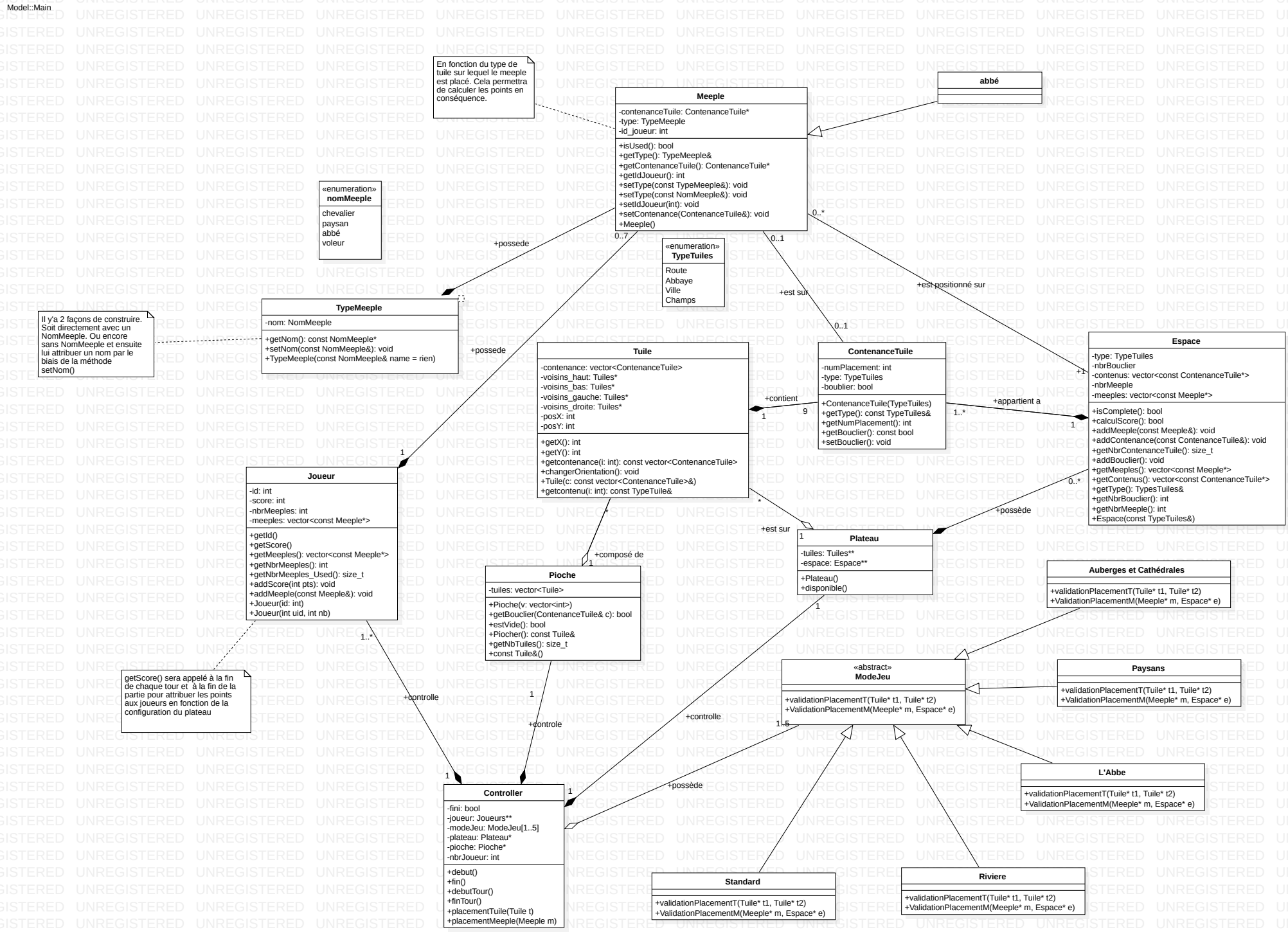
Model::Main


En fonction du type de tuile sur lequel le meeple est placé. Cela permettra de calculer les points en conséquence.
Meeple
-contenanceTuile: ContenanceTuile*
-type: TypeMeeple
-id_joueur: int
+isUsed(): bool
+getType(): TypeMeeple&
+getContenanceTuile(): ContenanceTuile*
+getIdJoueur(): int
+setType(const TypeMeeple&): void
+setType(const NomMeeple&): void
+setIdJoueur(int): void
+setContenance(ContenanceTuile&): void
+Meeple()
abbé
«enumeration»
nomMeeple
chevalier
paysan
abbé
voleur
«enumeration»
TypeTuiles
Route
Abbaye
Ville
Champs
+possede
0..7
0..1
0..*
+est sur
0..1
+est positionné sur
+1
Il y'a 2 façons de construire. Soit directement avec un NomMeeple. Ou encore sans NomMeeple et ensuite lui attribuer nom par le biais de la méthode setNom()
TypeMeeple
-nom: NomMeeple
+getNom(): const NomMeeple*
+setNom(const NomMeeple&): void
+TypeMeeple(const NomMeeple& name = rien)
+possede
Tuile
-contenance: vector<ContenanceTuile>
-voisins_haut: Tuiles*
-voisins_bas: Tuiles*
-voisins_gauche: Tuiles*
-voisins_droite: Tuiles*
-posX: int
-posY: int
+getX(): int
+getY(): int
+getcontenance(i: int): const vector<ContenanceTuile>
+changerOrientation(): void
+Tuile(c: const vector<ContenanceTuile>&)
+getcontenu(i: int): const TypeTuile&
+contient
1
9
ContenanceTuile
-numPlacement: int
-type: TypeTuiles
-boublier: bool
+ContenanceTuile(TypeTuiles)
+getType(): const TypeTuiles&
+getNumPlacement(): int
+getBouclier(): const bool
+setBouclier(): void
+appartient a
1..*
1
Espace
-type: TypeTuiles
-nbrBouclier
-contenus: vector<const ContenanceTuile*>
-nbrMeeple
-meeples: vector<const Meeple*>
+isComplete(): bool
+calculScore(): bool
+addMeeple(const Meeple&): void
+addContenance(const ContenanceTuile&): void
+getNbrContenanceTuile(): size_t
+addBouclier(): void
+getMeeples(): vector<const Meeple*>
+getContenus(): vector<const ContenanceTuile*>
+getType(): TypesTuiles&
+getNbrBouclier(): int
+getNbrMeeple(): int
+Espace(const TypeTuiles&)
1
Joueur
-id: int
-score: int
-nbrMeeples: int
-meeples: vector<const Meeple*>
+getId()
+getScore()
+getMeeples(): vector<const Meeple*>
+getNbrMeeples(): int
+getNbrMeeples_Used(): size_t
+addScore(int pts): void
+addMeeple(const Meeple&): void
+Joueur(id: int)
+Joueur(int uid, int nb)
*
+composé de
1
*
+est sur
0..*
+possède
1
Pioche
-tuiles: vector<Tuile>
+Pioche(v: vector<int>)
+getBouclier(ContenanceTuile& c): bool
+estVide(): bool
+Piocher(): const Tuile&
+getNbTuiles(): size_t
+const Tuile&()
Plateau
-tuiles: Tuiles**
-espace: Espace**
+Plateau()
+disponible()
1
Auberges et Cathédrales
+validationPlacementT(Tuile* t1, Tuile* t2)
+ValidationPlacementM(Meeple* m, Espace* e)
1..*
+controlle
1
+controle
1
getScore() sera appelé à la fin de chaque tour et à la fin de la partie pour attribuer les points aux joueurs en fonction de la configuration du plateau
«abstract»
ModeJeu
+validationPlacementT(Tuile* t1, Tuile* t2)
+ValidationPlacementM(Meeple* m, Espace* e)
Paysans
+validationPlacementT(Tuile* t1, Tuile* t2)
+ValidationPlacementM(Meeple* m, Espace* e)
+controlle
1..5
1
Controller
-fini: bool
-joueur: Joueurs**
-modeJeu: ModeJeu[1..5]
-plateau: Plateau*
-pioche: Pioche*
-nbrJoueur: int
+debut()
+fin()
+debutTour()
+finTour()
+placementTuile(Tuile t)
+placementMeeple(Meeple m)
+possède
L'Abbe
+validationPlacementT(Tuile* t1, Tuile* t2)
+ValidationPlacementM(Meeple* m, Espace* e)
Standard
+validationPlacementT(Tuile* t1, Tuile* t2)
+ValidationPlacementM(Meeple* m, Espace* e)
Riviere
+validationPlacementT(Tuile* t1, Tuile* t2)
+ValidationPlacementM(Meeple* m, Espace* e)