

# UML

## 1) Coté serveur (modele)

UserShema
+pseudo +password +key +nom +prenom +nombreCombatGagne +nombreCombatPerdu +tauxDeGain +dateDerniereConexion

## 2) Coté client (prototypes)

chatGlobal
+monPseudo +socket
+emitMessage() +getMessage() +disignMessage()

playerProto
+sprite +rectLife +fireBall +fireBallMoving +live +tailleDeplacementLateral +pseudo +moi +etat +perso +canMove +canAttac

+playerMooving -that +direction +idCombat
+setLife() +setDirection() +setEtat() +setImage() +changerImageDeplacement() +moveLeft() +moveRight() +sAbaissier() +seRelever() +emitUpdate() +copy() +handPunsh() +stopHandPunsh() -canAttacAgain() +stopFootPunsh() +fireBallAttac() -fireBallSendAttac() -endFireBallAttac() +movingFireBall() +bloc() +unBloc() +jump() -decende() -vitesse() +jAiGagne()

### 3) Notes

Le modèle du point 1 sert à la base de donnée mongoDB. Mais il est également repris coté client pour les objets d'AngularJS sur la page « ladder ». (sur cette page, les mots de passes ne sont pas présents pour des raisons évidentes de sécurité).

Le chat global est celui de la page « home ». Celui de la page de combat n'est pas un prototype et fonctionne légèrement différemment.

Le PlayerProto sert aux combat.

Si vous avez la moindre question, n'hésitez pas à nous recontacter. Le code est long et complexe, et nous le savons.