

H₂S

1.0

Généré par Doxygen 1.8.14

Table des matières

1	Page Principale	1
2	Liste des choses à faire	3
3	Index hiérarchique	5
3.1	Hiérarchie des classes	5
4	Index des structures de données	7
4.1	Structures de données	7
5	Index des fichiers	9
5.1	Liste des fichiers	9
6	Documentation des structures de données	11
6.1	Référence de la classe Att_soi	12
6.1.1	Description détaillée	13
6.1.2	Documentation des fonctions membres	13
6.1.2.1	lancerUnSoin()	13
6.2	Référence de la classe Base_personnage	13
6.2.1	Description détaillée	15
6.2.2	Documentation des constructeurs et destructeur	15
6.2.2.1	__construct()	15
6.2.3	Documentation des fonctions membres	16
6.2.3.1	att()	16
6.2.3.2	etoile()	16
6.2.3.3	hydrate()	16

6.2.3.4	id_persos()	17
6.2.3.5	nom()	17
6.2.3.6	pv()	17
6.2.3.7	qualite()	18
6.2.3.8	setAtt()	18
6.2.3.9	setEmp_team()	18
6.2.3.10	setId_persos()	19
6.2.3.11	setNom()	19
6.2.3.12	setPv()	19
6.2.3.13	setQualite()	20
6.2.3.14	setStuff_1()	20
6.2.3.15	setStuff_2()	21
6.2.3.16	setStuff_3()	21
6.2.3.17	setStuff_4()	21
6.2.3.18	setTypes()	22
6.2.3.19	stuff_1()	22
6.2.3.20	stuff_2()	22
6.2.3.21	stuff_3()	23
6.2.3.22	stuff_4()	23
6.2.3.23	types()	23
6.2.4	Documentation des champs	24
6.2.4.1	\$att	24
6.2.4.2	\$emp_team	24
6.2.4.3	\$etoile	24
6.2.4.4	\$id_persos	24
6.2.4.5	\$nom	24
6.2.4.6	\$pv	25
6.2.4.7	\$qualite	25
6.2.4.8	\$stuff_1	25
6.2.4.9	\$stuff_2	25

6.2.4.10	\$stuff_3	25
6.2.4.11	\$stuff_4	26
6.2.4.12	\$types	26
6.3	Référence de la classe Personnage	26
6.3.1	Description détaillée	29
6.3.2	Documentation des constructeurs et destructeur	29
6.3.2.1	__construct()	29
6.3.3	Documentation des fonctions membres	29
6.3.3.1	arm()	30
6.3.3.2	atout()	30
6.3.3.3	crit()	31
6.3.3.4	debut_combat()	31
6.3.3.5	def()	31
6.3.3.6	esc()	32
6.3.3.7	estEndormi()	32
6.3.3.8	etat()	33
6.3.3.9	etoile()	33
6.3.3.10	frapper()	34
6.3.3.11	hydrate()	34
6.3.3.12	id_perso()	35
6.3.3.13	id_user()	35
6.3.3.14	incrementAtout()	36
6.3.3.15	niveau()	36
6.3.3.16	pen()	36
6.3.3.17	pre()	37
6.3.3.18	qualite()	37
6.3.3.19	recevoirDegats()	38
6.3.3.20	recevoirSoins()	38
6.3.3.21	recevoirXP()	38
6.3.3.22	veille()	39

6.3.3.23	setArm()	40
6.3.3.24	setAtout()	40
6.3.3.25	setCrit()	40
6.3.3.26	setDef()	41
6.3.3.27	setEsc()	41
6.3.3.28	setEtat()	42
6.3.3.29	setEtoile()	42
6.3.3.30	setId_perso()	42
6.3.3.31	setId_user()	43
6.3.3.32	setNiveau()	43
6.3.3.33	setPen()	44
6.3.3.34	setPre()	44
6.3.3.35	setQualite()	44
6.3.3.36	setSoi()	45
6.3.3.37	setTeam()	45
6.3.3.38	setTen()	46
6.3.3.39	setTimeEndormi()	46
6.3.3.40	setTypes()	46
6.3.3.41	setVit()	47
6.3.3.42	setXp()	47
6.3.3.43	setXp_max()	48
6.3.3.44	soi()	48
6.3.3.45	soigner()	48
6.3.3.46	team()	49
6.3.3.47	ten()	50
6.3.3.48	timeEndormi()	50
6.3.3.49	types()	50
6.3.3.50	vit()	51
6.3.3.51	xp()	51
6.3.3.52	xp_max()	52

6.3.4	Documentation des champs	52
6.3.4.1	\$arm	52
6.3.4.2	\$atout	52
6.3.4.3	\$crit	52
6.3.4.4	\$def	53
6.3.4.5	\$esc	53
6.3.4.6	\$etat	53
6.3.4.7	\$etoile	53
6.3.4.8	\$id_perso	53
6.3.4.9	\$id_user	53
6.3.4.10	\$niveau	54
6.3.4.11	\$pen	54
6.3.4.12	\$pre	54
6.3.4.13	\$qualite	54
6.3.4.14	\$soi	54
6.3.4.15	\$team	54
6.3.4.16	\$ten	55
6.3.4.17	\$timeEndormi	55
6.3.4.18	\$vit	55
6.3.4.19	\$xp	55
6.3.4.20	\$xp_max	55
6.3.4.21	CEST_MOI	55
6.3.4.22	PAS_DE_MAGIE	56
6.3.4.23	PERSO_ENDORMI	56
6.3.4.24	PERSO_FULL_LIFE	56
6.3.4.25	PERSONNAGE_ENSORCELE	56
6.3.4.26	PERSONNAGE_FRAPPE	56
6.3.4.27	PERSONNAGE_SOIGNE	56
6.3.4.28	PERSONNAGE_TUE	57
6.4	Référence de la classe PersonnagesManager	57

6.4.1	Description détaillée	57
6.4.2	Documentation des constructeurs et destructeur	57
6.4.2.1	__construct()	57
6.4.3	Documentation des fonctions membres	58
6.4.3.1	add()	58
6.4.3.2	count()	58
6.4.3.3	delete()	58
6.4.3.4	exists()	58
6.4.3.5	get()	58
6.4.3.6	get_base_perso()	59
6.4.3.7	get_id_team()	59
6.4.3.8	getList()	59
6.4.3.9	getList_up()	59
6.4.3.10	getListCheat()	59
6.4.3.11	setDb()	60
6.4.3.12	sup_team()	60
6.4.3.13	update()	60
6.4.3.14	update_team()	60
6.5	Référence de la classe Soigneur	61
6.5.1	Description détaillée	62
6.5.2	Documentation des fonctions membres	62
6.5.2.1	lancerUnSoin()	62
6.6	Référence de la classe Stuff	62
6.6.1	Description détaillée	63
6.6.2	Documentation des constructeurs et destructeur	63
6.6.2.1	__construct()	63
6.6.3	Documentation des fonctions membres	63
6.6.3.1	atr_1()	64
6.6.3.2	atr_2()	64
6.6.3.3	cout()	65

6.6.3.4	descr()	65
6.6.3.5	etoile()	65
6.6.3.6	hydrate()	66
6.6.3.7	id_base()	66
6.6.3.8	id_stuff()	66
6.6.3.9	lvlUp()	67
6.6.3.10	niveau()	67
6.6.3.11	nom()	68
6.6.3.12	qualite()	68
6.6.3.13	setAtr_1()	68
6.6.3.14	setAtr_2()	69
6.6.3.15	setCout()	69
6.6.3.16	setDescr()	69
6.6.3.17	setEtoile()	70
6.6.3.18	setId_base()	70
6.6.3.19	setId_stuff()	71
6.6.3.20	setNiveau()	71
6.6.3.21	setNom()	71
6.6.3.22	setQualite()	72
6.6.3.23	setTypes()	72
6.6.3.24	types()	73
6.6.4	Documentation des champs	73
6.6.4.1	\$atr_1	73
6.6.4.2	\$atr_2	73
6.6.4.3	\$cout	73
6.6.4.4	\$descr	74
6.6.4.5	\$etoile	74
6.6.4.6	\$id_base	74
6.6.4.7	\$id_stuff	74
6.6.4.8	\$niveau	74

6.6.4.9	\$nom	74
6.6.4.10	\$qualite	75
6.6.4.11	\$types	75
6.7	Référence de la classe StuffManager	75
6.7.1	Description détaillée	75
6.7.2	Documentation des constructeurs et destructeur	75
6.7.2.1	__construct()	76
6.7.3	Documentation des fonctions membres	76
6.7.3.1	add()	76
6.7.3.2	count()	76
6.7.3.3	delete()	76
6.7.3.4	exists()	77
6.7.3.5	get()	77
6.7.3.6	get_base_perso()	77
6.7.3.7	get_niveau_sup()	77
6.7.3.8	getList()	77
6.7.3.9	getListCheat()	77
6.7.3.10	setDb()	78
6.7.3.11	sup_team()	78
6.7.3.12	update()	78
6.7.3.13	update_team()	78
6.8	Référence de la classe User	79
6.8.1	Description détaillée	79
6.8.2	Documentation des constructeurs et destructeur	79
6.8.2.1	__construct()	80
6.8.3	Documentation des fonctions membres	80
6.8.3.1	argent()	80
6.8.3.2	email()	81
6.8.3.3	emp_team_max()	81
6.8.3.4	endu()	81

6.8.3.5	endu_Max()	82
6.8.3.6	hydrate()	82
6.8.3.7	id_user()	82
6.8.3.8	mdp()	83
6.8.3.9	niveau()	83
6.8.3.10	nomValide()	83
6.8.3.11	or_()	84
6.8.3.12	payer_argent()	84
6.8.3.13	pseudo()	84
6.8.3.14	setArgent()	85
6.8.3.15	setEmail()	85
6.8.3.16	setEmp_team_max()	85
6.8.3.17	setEndu()	86
6.8.3.18	setEndu_Max()	86
6.8.3.19	setId_user()	87
6.8.3.20	setMdp()	87
6.8.3.21	setNiveau()	87
6.8.3.22	setOr_()	88
6.8.3.23	setPseudo()	88
6.8.3.24	setXp()	89
6.8.3.25	setXp_Max()	89
6.8.3.26	xp()	89
6.8.3.27	xp_Max()	90
6.8.4	Documentation des champs	90
6.8.4.1	\$argent	90
6.8.4.2	\$email	90
6.8.4.3	\$emp_team_max	90
6.8.4.4	\$endu	90
6.8.4.5	\$endu_Max	90
6.8.4.6	\$id_user	91

6.8.4.7	\$mdp	91
6.8.4.8	\$niveau	91
6.8.4.9	\$or_	91
6.8.4.10	\$pseudo	91
6.8.4.11	\$xp	91
6.8.4.12	\$Xp_Max	92
6.9	Référence de la classe UserManager	92
6.9.1	Description détaillée	92
6.9.2	Documentation des constructeurs et destructeur	92
6.9.2.1	__construct()	92
6.9.3	Documentation des fonctions membres	93
6.9.3.1	add()	93
6.9.3.2	count()	93
6.9.3.3	delete()	93
6.9.3.4	exists()	93
6.9.3.5	get()	93
6.9.3.6	getList()	94
6.9.3.7	setDb()	94
6.9.3.8	update()	94
6.9.3.9	verifMdp()	94

7	Documentation des fichiers	95
7.1	Référence du fichier ajax/add_perso.php	95
7.1.1	Documentation des fonctions	95
7.1.1.1	chargerClasse()	95
7.1.2	Documentation des variables	95
7.1.2.1	\$manager	96
7.1.2.2	\$managerStuff	96
7.1.2.3	\$perso	96
7.1.2.4	\$user	96
7.1.2.5	\$usermanager	96
7.2	Référence du fichier ajax/gestion_perso.php	96
7.2.1	Documentation des variables	97
7.2.1.1	else	97
7.3	Référence du fichier ajax/gestion_stuff.php	97
7.3.1	Documentation des fonctions	97
7.3.1.1	chargerClasse()	97
7.3.2	Documentation des variables	97
7.3.2.1	\$manager	98
7.3.2.2	\$managerStuff	98
7.3.2.3	\$managerUser	98
7.3.2.4	\$perso	98
7.3.2.5	\$stuff	98
7.3.2.6	\$tab	98
7.3.2.7	else	99
7.4	Référence du fichier ajax/gestion_sup_tab.php	99
7.5	Référence du fichier ajax/gestion_up_perso.php	99
7.5.1	Documentation des fonctions	99
7.5.1.1	chargerClasse()	99
7.5.2	Documentation des variables	99
7.5.2.1	\$entry	100

7.5.2.2	\$json	100
7.5.2.3	\$manager	100
7.5.2.4	\$perso	100
7.6	Référence du fichier ajax/sup_perso.php	100
7.6.1	Documentation des fonctions	101
7.6.1.1	chargerClasse()	101
7.6.2	Documentation des variables	101
7.6.2.1	\$manager	101
7.6.2.2	\$perso	101
7.7	Référence du fichier ajax/team_ajax.php	101
7.7.1	Documentation des fonctions	102
7.7.1.1	chargerClasse()	102
7.7.2	Documentation des variables	102
7.7.2.1	\$manager	102
7.8	Référence du fichier class/Att_soi.php	102
7.9	Référence du fichier class/Base_personnage.php	102
7.9.1	Description détaillée	102
7.10	Référence du fichier class/Personnage.php	103
7.10.1	Description détaillée	103
7.11	Référence du fichier class/PersonnagesManager.php	103
7.12	Référence du fichier class/Soigneur.php	103
7.13	Référence du fichier class/Stuff.php	103
7.14	Référence du fichier includes/stuff.php	104
7.14.1	Description détaillée	104
7.15	Référence du fichier class/StuffManager.php	104
7.16	Référence du fichier class/User.php	104
7.17	Référence du fichier class/UserManager.php	104
7.18	Référence du fichier includes/cheat.php	104
7.18.1	Description détaillée	105
7.19	Référence du fichier includes/cheat/modif_stuff.php	105

7.19.1 Description détaillée	105
7.20 Référence du fichier includes/cheat/modif_team.php	105
7.20.1 Description détaillée	106
7.21 Référence du fichier includes/eviterMessageAvertissement.php	106
7.22 Référence du fichier includes/Function_Attaque.php	106
7.22.1 Description détaillée	106
7.23 Référence du fichier includes/Function_Jeu.php	106
7.23.1 Description détaillée	107
7.23.2 Documentation des fonctions	107
7.23.2.1 affichePerso()	107
7.23.2.2 fonctionComparaison()	108
7.24 Référence du fichier includes/identifiant.php	108
7.24.1 Description détaillée	108
7.25 Référence du fichier includes/stuff_modal.php	108
7.25.1 Description détaillée	108
7.26 Référence du fichier includes/team - save.php	109
7.27 Référence du fichier includes/team.php	109
7.27.1 Description détaillée	109
7.27.2 Documentation des fonctions	109
7.27.2.1 cadre_team()	109
7.28 Référence du fichier includes/up_perso.php	110
7.28.1 Description détaillée	110
7.29 Référence du fichier includes/up_perso_modal.php	110
7.29.1 Description détaillée	110
7.30 Référence du fichier index.php	111
7.30.1 Description détaillée	111
7.30.2 Documentation des fonctions	111
7.30.2.1 chargerClass()	111
7.31 Référence du fichier JS/add_perso.js	112
7.31.1 Description détaillée	112

7.31.2	Documentation des fonctions	112
7.31.2.1	auto_load()	112
7.31.2.2	submit()	113
7.32	Référence du fichier JS/highlight/highlight.pack.js	113
7.32.1	Documentation des variables	115
7.32.1.1	css	115
7.32.1.2	hljs	116
7.32.1.3	html	117
7.32.1.4	HTML_ATTR	117
7.32.1.5	HTML_DOCTYPE	118
7.32.1.6	HTML_SHORT_ATTR	118
7.32.1.7	HTML_TAGS	118
7.32.1.8	HTML_VALUE	118
7.32.1.9	initHighlightingOnLoad	118
7.32.1.10	javascript	119
7.32.1.11	xml	119
7.32.1.12	XML_ATTR	119
7.32.1.13	XML_COMMENT	119
7.32.1.14	XML_VALUE_APOS	119
7.32.1.15	XML_VALUE_QUOT	120
7.33	Référence du fichier JS/jquery-Rotate-Plugin.js	120
7.33.1	Documentation des variables	120
7.33.1.1	rotate	120
7.34	Référence du fichier JS/jquery.easing.1.3.js	120
7.34.1	Documentation des fonctions	121
7.34.1.1	extend()	121
7.34.2	Documentation des variables	122
7.34.2.1	easing	122
7.35	Référence du fichier JS/jquery.modal.js	122
7.35.1	Documentation des fonctions	122
7.35.1.1	function() [1/2]	122
7.35.1.2	function() [2/2]	123
7.36	Référence du fichier JS/sup_perso.js	123
7.36.1	Description détaillée	123
7.36.2	Documentation des fonctions	123
7.36.2.1	auto_load()	123
7.36.2.2	submit()	124
7.37	Référence du fichier JS/team.js	124
7.37.1	Description détaillée	124
7.37.2	Documentation des fonctions	124
7.37.2.1	auto_load()	124
7.37.2.2	submit()	124

Chapitre 1

Page Principale

Jeu en construction. Jeu de cartes à collectionner, constitution d'un deck, combats PVE et PVP automatiques.
Schéma BDD : Abres d'inclusion des fichiers php :

```
Arbre d'inclusions :
- index.php
  |- ./class/.php
  |- ./includes/identifiant.php
  |- ./includes/Function_Jeu.php
  |- ./includes/cheat.php
  |   |- ./includes/cheat/modif_team.php
  |   |- ./includes/cheat/modif_stuff.php
  |- ./includes/team.php
  |- ./includes/stuff.php
  |- ./includes/up_perso.php
9 fichiers inclus
```


Chapitre 2

Liste des choses à faire

Fichier [index.php](#)

Il reste à implémenter les scripts de combats PVE et PVP.

Fichier [modif_stuff.php](#)

Finir l'interface

Fichier [stuff_modal.php](#)

Implémenter la fonctionnalité pour up plusieurs niveaux

Chapitre 3

Index hiérarchique

3.1 Hiérarchie des classes

Cette liste d'héritage est classée approximativement par ordre alphabétique :

Base_personnage	13
Personnage	26
Att_soi	12
Seigneur	61
PersonnagesManager	57
Stuff	62
StuffManager	75
User	79
UserManager	92

Chapitre 4

Index des structures de données

4.1 Structures de données

Liste des structures de données avec une brève description :

Att_soi	12
Base_personnage	13
Personnage	26
PersonnagesManager	57
Soigneur	61
Stuff	62
StuffManager	75
User	79
UserManager	92

Chapitre 5

Index des fichiers

5.1 Liste des fichiers

Liste de tous les fichiers avec une brève description :

index.php	
Page d'index du jeu	111
ajax/add_perso.php	95
ajax/gestion_perso.php	96
ajax/gestion_stuff.php	97
ajax/gestion_sup_tab.php	99
ajax/gestion_up_perso.php	99
ajax/sup_perso.php	100
ajax/team_ajax.php	101
class/Att_soi.php	102
class/Base_personnage.php	
Class de base des personnages	102
class/Personnage.php	
Class des personnages	103
class/PersonnagesManager.php	103
class/Soigneur.php	103
class/Stuff.php	103
class/StuffManager.php	104
class/User.php	104
class/UserManager.php	104
includes/cheat.php	
Menu cheat du jeu	104
includes/eviterMessageAvertissement.php	106
includes/Function_Attaque.php	
Définit les fonctions d'un combat	106
includes/Function_Jeu.php	
Définit les fonctions du jeu	106
includes/identifiant.php	
Connexion PDO	108
includes/stuff.php	
Menu de gestion du stuff	104
includes/stuff_modal.php	
Modal de gestion d'un équipement	108
includes/team_save.php	109
includes/team.php	
Menu de gestion d'équipe	109

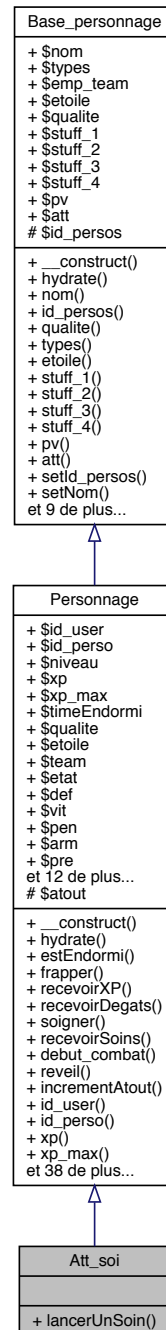
includes/ up_perso.php	
Menu d'amélioration des personnages	110
includes/ up_perso_modal.php	
Modal permettant la selection des personnages à sacrifier pour en up un autre	110
includes/cheat/ modif_stuff.php	
Menu cheat du stuff	105
includes/cheat/ modif_team.php	
Menu cheat de la team	105
JS/ add_perso.js	
Ajax pour l'ajout' de personnage	112
JS/ jQuery-Rotate-Plugin.js	120
JS/ jquery.easing.1.3.js	120
JS/ jquery.modal.js	122
JS/ sup_perso.js	
Ajax pour la suppression de personnage	123
JS/ team.js	
Ajax pour gestion de l'équipe	124
JS/highlight/ highlight.pack.js	113

Chapitre 6

Documentation des structures de données

6.1 Référence de la classe Att_soi

Graphe d'héritage de Att_soi:



Fonctions membres publiques

— `lancerUnSoin` (`Personnage $perso`)

Membres hérités additionnels

6.1.1 Description détaillée

Définition à la ligne 3 du fichier Att_soi.php.

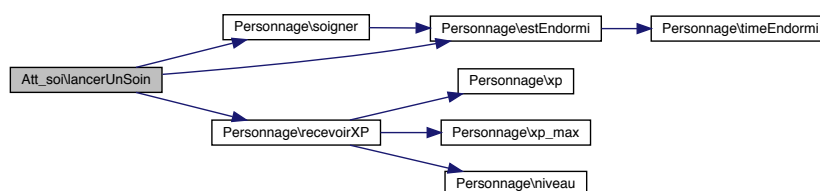
6.1.2 Documentation des fonctions membres

6.1.2.1 lancerUnSoin()

```
Att_soi::lancerUnSoin (
    Personnage $perso )
```

Définition à la ligne 5 du fichier Att_soi.php.

Voici le graphe d'appel pour cette fonction :



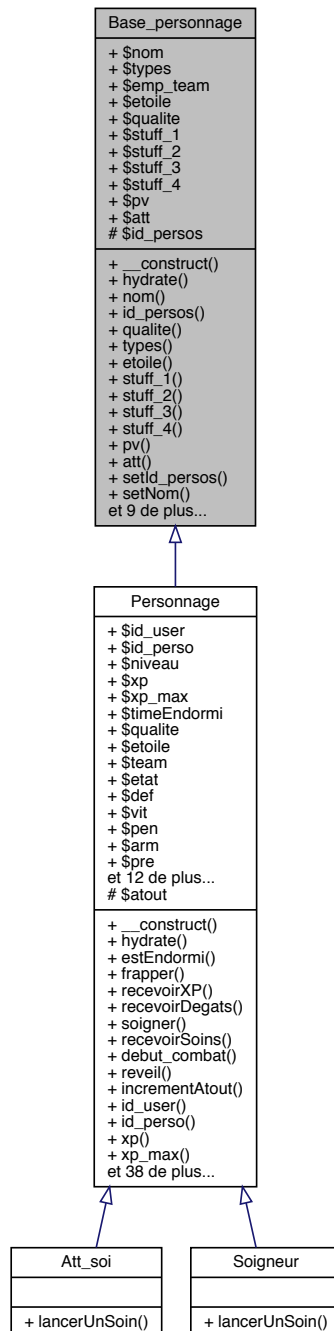
La documentation de cette classe a été générée à partir du fichier suivant :

— class/[Att_soi.php](#)

6.2 Référence de la classe Base_personnage

```
#include "class/Base_personnage.php"
```

Graphe d'héritage de Base_personnage:



Fonctions membres publiques

- `__construct` (\$valeurs=array())
- `hydrate` (array \$donnees)
- `nom` ()
- `id_persos` ()
- `qualite` ()
- `types` ()
- `etoile` ()


```

— stuff_1 ()
— stuff_2 ()
— stuff_3 ()
— stuff_4 ()
— pv ()
— att ()
— setId_persos ($id_persos)
— setNom ($nom)
— setQualite ($qualite)
— setTypes ($types)
— setPv ($pv)
— setAtt ($att)
— setEmp_team ($emp_team)
— setStuff_1 ($stuff_1)
— setStuff_2 ($stuff_2)
— setStuff_3 ($stuff_3)
— setStuff_4 ($stuff_4)

```

Champs de données

```

— $nom
— $types
— $emp_team
— $etoile
— $qualite
— $stuff_1
— $stuff_2
— $stuff_3
— $stuff_4
— $pv
— $att

```

Attributs protégés

```

— $id_persos

```

6.2.1 Description détaillée

Définition à la ligne 13 du fichier Base_personnage.php.

6.2.2 Documentation des constructeurs et destructeur

6.2.2.1 __construct()

```

Base_personnage::__construct (
    $valeurs = array() )

```

Définition à la ligne 27 du fichier Base_personnage.php.

Voici le graphe d'appel pour cette fonction :



6.2.3 Documentation des fonctions membres

6.2.3.1 att()

```
Base_personnage::att ( )
```

Définition à la ligne 63 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.2 etoile()

```
Base_personnage::etoile ( )
```

Définition à la ligne 57 du fichier Base_personnage.php.

6.2.3.3 hydrate()

```
Base_personnage::hydrate (
    array $donnees )
```

Définition à la ligne 35 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.4 id_persos()

`Base_personnage::id_persos ()`

Définition à la ligne 54 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.5 nom()

`Base_personnage::nom ()`

Définition à la ligne 53 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.6 pv()

`Base_personnage::pv ()`

Définition à la ligne 62 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.7 qualite()

```
Base_personnage::qualite ( )
```

Définition à la ligne 55 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.8 setAtt()

```
Base_personnage::setAtt (
    $att )
```

Définition à la ligne 114 du fichier Base_personnage.php.

Voici le graphe d'appel pour cette fonction :



6.2.3.9 setEmp_team()

```
Base_personnage::setEmp_team (
    $emp_team )
```

Définition à la ligne 125 du fichier Base_personnage.php.

6.2.3.10 setId_persos()

```
Base_personnage::setId_persos (
    $id_persos )
```

Définition à la ligne 67 du fichier Base_personnage.php.

Voici le graphe d'appel pour cette fonction :



6.2.3.11 setNom()

```
Base_personnage::setNom (
    $nom )
```

Définition à la ligne 73 du fichier Base_personnage.php.

Voici le graphe d'appel pour cette fonction :



6.2.3.12 setPv()

```
Base_personnage::setPv (
    $pv )
```

Définition à la ligne 103 du fichier Base_personnage.php.

Voici le graphe d'appel pour cette fonction :



6.2.3.13 `setQualite()`

```
Base_personnage::setQualite (
    $qualite )
```

Définition à la ligne 83 du fichier `Base_personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.2.3.14 `setStuff_1()`

```
Base_personnage::setStuff_1 (
    $stuff_1 )
```

Définition à la ligne 136 du fichier `Base_personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.2.3.15 setStuff_2()

```
Base_personnage::setStuff_2 (
    $stuff_2 )
```

Définition à la ligne 147 du fichier Base_personnage.php.

Voici le graphe d'appel pour cette fonction :



6.2.3.16 setStuff_3()

```
Base_personnage::setStuff_3 (
    $stuff_3 )
```

Définition à la ligne 158 du fichier Base_personnage.php.

Voici le graphe d'appel pour cette fonction :



6.2.3.17 setStuff_4()

```
Base_personnage::setStuff_4 (
    $stuff_4 )
```

Définition à la ligne 169 du fichier Base_personnage.php.

Voici le graphe d'appel pour cette fonction :



6.2.3.18 setTypes()

```
Base_personnage::setTypes (
    $types )
```

Définition à la ligne 93 du fichier Base_personnage.php.

Voici le graphe d'appel pour cette fonction :



6.2.3.19 stuff_1()

```
Base_personnage::stuff_1 ( )
```

Définition à la ligne 58 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.20 stuff_2()

```
Base_personnage::stuff_2 ( )
```

Définition à la ligne 59 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.21 stuff_3()

```
Base_personnage::stuff_3 ( )
```

Définition à la ligne 60 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.22 stuff_4()

```
Base_personnage::stuff_4 ( )
```

Définition à la ligne 61 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.3.23 types()

```
Base_personnage::types ( )
```

Définition à la ligne 56 du fichier Base_personnage.php.

Voici le graphe des appelants de cette fonction :



6.2.4 Documentation des champs

6.2.4.1 \$att

`Base_personnage::$att`

degat du personnage

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.2 \$emp_team

`Base_personnage::$emp_team`

emplacement si dans l'équipe

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.3 \$etoile

`Base_personnage::$etoile`

étoile du personnage

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.4 \$id_persos

`Base_personnage::$id_persos` [protected]

id du personnage

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.5 \$nom

`Base_personnage::$nom`

nom du personnage

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.6 \$pv

```
Base_personnage::$pv
```

point de vie du personnage

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.7 \$qualite

```
Base_personnage::$qualite
```

qualité du personnage

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.8 \$stuff_1

```
Base_personnage::$stuff_1
```

id de l'équipement n°1

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.9 \$stuff_2

```
Base_personnage::$stuff_2
```

id de l'équipement n°2

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.10 \$stuff_3

```
Base_personnage::$stuff_3
```

id de l'équipement n°3

Définition à la ligne 14 du fichier Base_personnage.php.

6.2.4.11 \$stuff_4

`Base_personnage::$stuff_4`

id de l'équipement n°4 , vide par default

Définition à la ligne 14 du fichier `Base_personnage.php`.

6.2.4.12 \$types

`Base_personnage::$types`

type du personnage

Définition à la ligne 14 du fichier `Base_personnage.php`.

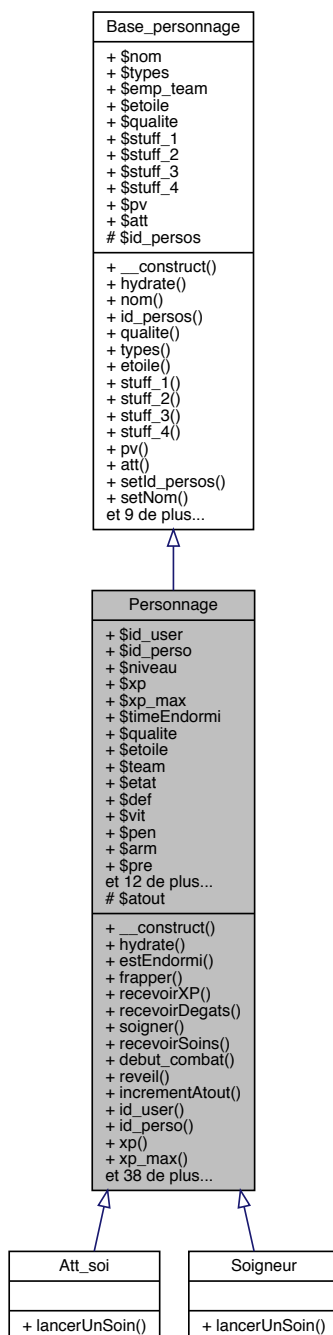
La documentation de cette classe a été générée à partir du fichier suivant :

— [class/Base_personnage.php](#)

6.3 Référence de la classe Personnage

```
#include "class/Personnage.php"
```

Graphe d'héritage de Personnage:



Fonctions membres publiques

- `__construct` (\$valeurs=array())
- `hydrate` (array \$donnees)
- `estEndormi` ()
- `frapper` (Personnage \$perso, Personnage \$perso2)
Fonction pour frapper.
- `recevoirXP` (\$val)

```

— recevoirDegats ($perso2)
— soigner (Personnage $perso)
— recevoirSoins ($perso)
— debut_combat ()
— reveil ()
— incrementAtout ()
— id_user ()
— id_perso ()
— xp ()
— xp_max ()
— niveau ()
— qualite ()
— etoile ()
— atout ()
— timeEndormi ()
— types ()
— team ()
— def ()
— vit ()
— pen ()
— arm ()
— pre ()
— esc ()
— crit ()
— ten ()
— soi ()
— etat ()
— setAtout ($atout)
— setId_user ($id_user)
— setId_perso ($id_perso)
— setEtoile ($etoile)
— setXp ($xp)
— setXp_max ($xp_max)
— setQualite ($qualite)
— setNiveau ($niveau)
— setTimeEndormi ($time)
— setTypes ($types)
— setTeam ($team)
— setDef ($def)
— setVit ($vit)
— setPen ($pen)
— setArm ($arm)
— setPre ($pre)
— setEsc ($esc)
— setCrit ($crit)
— setTen ($ten)
— setSoi ($soi)
— setEtat ($etat)

```

Champs de données

```

— $id_user
— $id_perso
— $niveau
— $xp
— $xp_max
— $timeEndormi
— $qualite
— $etoile
— $team
— $etat
— $def
— $vit
— $pen
— $arm
— $pre
— $esc
— $crit
— $ten
— $soi
— const CEST_MOI = 1
— const PERSONNAGE_TUE = 2
— const PERSONNAGE_FRAPPE = 3
— const PERSONNAGE_ENSORCELE = 4
— const PAS_DE_MAGIE = 5

```

- const `PERSO_ENDORMI` = 6
- const `PERSONNAGE_SOIGNE` = 7
- const `PERSO_FULL_LIFE` = 8

Attributs protégés

- `$atout`

6.3.1 Description détaillée

Définition à la ligne 13 du fichier Personnage.php.

6.3.2 Documentation des constructeurs et destructeur

6.3.2.1 `__construct()`

```
Personnage::__construct (
    $valeurs = array() )
```

Définition à la ligne 45 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3 Documentation des fonctions membres

6.3.3.1 arm()

```
Personnage::arm ( )
```

Définition à la ligne 191 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :

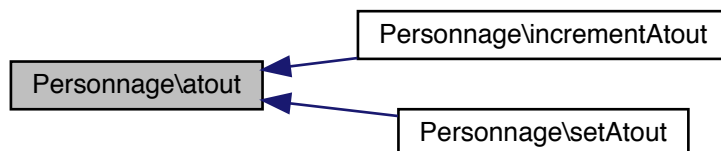


6.3.3.2 atout()

```
Personnage::atout ( )
```

Définition à la ligne 184 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :



6.3.3.3 crit()

```
Personnage::crit ( )
```

Définition à la ligne 194 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :

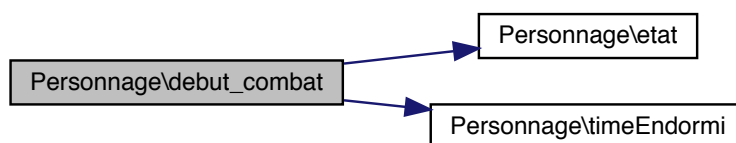


6.3.3.4 debut_combat()

```
Personnage::debut_combat ( )
```

Définition à la ligne 152 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.5 def()

```
Personnage::def ( )
```

Définition à la ligne 188 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :



6.3.3.6 `esc()`

```
Personnage::esc ( )
```

Définition à la ligne 193 du fichier `Personnage.php`.

Voici le graphe des appelants de cette fonction :



6.3.3.7 `estEndormi()`

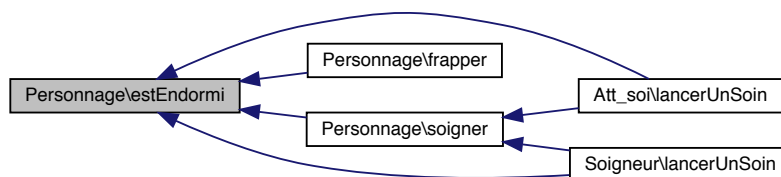
```
Personnage::estEndormi ( )
```

Définition à la ligne 69 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :

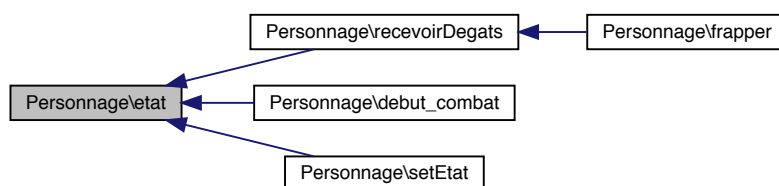


6.3.3.8 `etat()`

`Personnage::etat ()`

Définition à la ligne 197 du fichier `Personnage.php`.

Voici le graphe des appelants de cette fonction :



6.3.3.9 `etoile()`

`Personnage::etoile ()`

Définition à la ligne 183 du fichier `Personnage.php`.

Voici le graphe des appelants de cette fonction :



6.3.3.10 frapper()

```
Personnage::frapper (
    Personnage $perso,
    Personnage $perso2 )
```

Fonction pour frapper.

Fonction permettant à un personnage de frapper un autre personnage. S'il est sa propre cible ou s'il est endormi, il ne pourra pas frapper

Paramètres

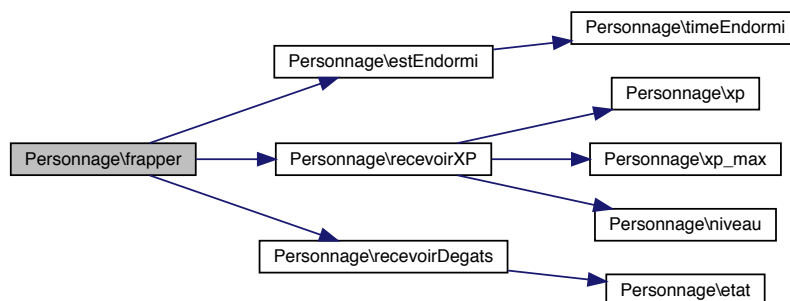
Personnage	\$perso	Perso qui se fait frapper.
Personnage	\$perso2	Person qui frappe.

Renvoie

Renvoie à la fonction qui permet de recevoir des dégats ("recevoirDegats()").

Définition à la ligne 82 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.11 hydrate()

```
Personnage::hydrate (
    array $donnees )
```

Définition à la ligne 53 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :



6.3.3.12 id_perso()

```
Personnage::id_perso ( )
```

Définition à la ligne 178 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :



6.3.3.13 id_user()

```
Personnage::id_user ( )
```

Définition à la ligne 177 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :

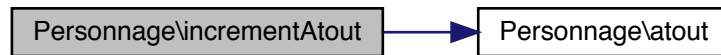


6.3.3.14 incrementAtout()

```
Personnage::incrementAtout ( )
```

Définition à la ligne 164 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :

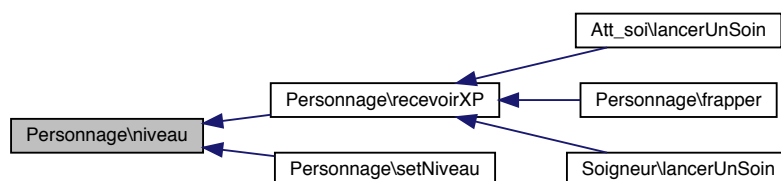


6.3.3.15 niveau()

```
Personnage::niveau ( )
```

Définition à la ligne 181 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :



6.3.3.16 pen()

```
Personnage::pen ( )
```

Définition à la ligne 190 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :



6.3.3.17 `pre()`

```
Personnage::pre ( )
```

Définition à la ligne 192 du fichier `Personnage.php`.

Voici le graphe des appelants de cette fonction :



6.3.3.18 `qualite()`

```
Personnage::qualite ( )
```

Définition à la ligne 182 du fichier `Personnage.php`.

Voici le graphe des appelants de cette fonction :



6.3.3.19 recevoirDegats()

```
Personnage::recevoirDegats (
    $perso2 )
```

Définition à la ligne 112 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



6.3.3.20 recevoirSoins()

```
Personnage::recevoirSoins (
    $perso )
```

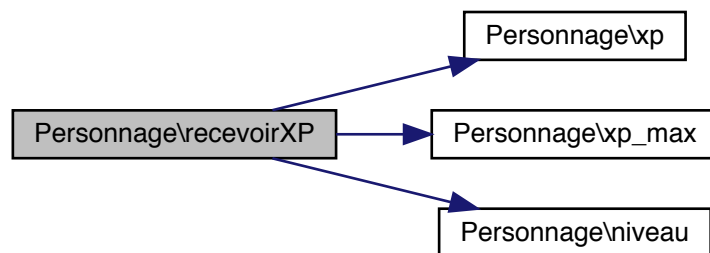
Définition à la ligne 141 du fichier Personnage.php.

6.3.3.21 recevoirXP()

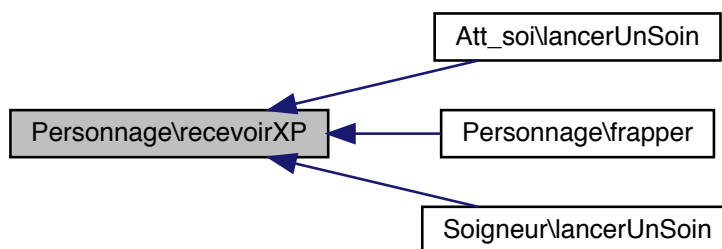
```
Personnage::recevoirXP (
    $val )
```

Définition à la ligne 100 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



6.3.3.22 `veille()`

`Personnage::veille ()`

Définition à la ligne 159 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.3.3.23 setArm()

```
Personnage::setArm (
    $arm )
```

Définition à la ligne 323 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.24 setAtout()

```
Personnage::setAtout (
    $atout )
```

Définition à la ligne 199 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.25 setCrit()

```
Personnage::setCrit (
    $crit )
```

Définition à la ligne 356 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.26 `setDef()`

```
Personnage::setDef (
    $def )
```

Définition à la ligne 290 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.3.3.27 `setEsc()`

```
Personnage::setEsc (
    $esc )
```

Définition à la ligne 345 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.3.3.28 setEtat()

```
Personnage::setEtat (
    $etat )
```

Définition à la ligne 389 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.29 setEtoile()

```
Personnage::setEtoile (
    $etoile )
```

Définition à la ligne 221 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.30 setId_perso()

```
Personnage::setId_perso (
    $id_perso )
```

Définition à la ligne 215 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.31 `setId_user()`

```
Personnage::setId_user (
    $id_user )
```

Définition à la ligne 209 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.3.3.32 `setNiveau()`

```
Personnage::setNiveau (
    $niveau )
```

Définition à la ligne 255 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.3.3.33 setPen()

```
Personnage::setPen (
    $pen )
```

Définition à la ligne 312 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.34 setPre()

```
Personnage::setPre (
    $pre )
```

Définition à la ligne 334 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.35 setQualite()

```
Personnage::setQualite (
    $qualite )
```

Définition à la ligne 245 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.36 `setSoi()`

```
Personnage::setSoi (
    $soi )
```

Définition à la ligne 378 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.3.3.37 `setTeam()`

```
Personnage::setTeam (
    $team )
```

Définition à la ligne 281 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.3.3.38 setTen()

```
Personnage::setTen (
    $ten )
```

Définition à la ligne 367 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.39 setTimeEndormi()

```
Personnage::setTimeEndormi (
    $time )
```

Définition à la ligne 266 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :

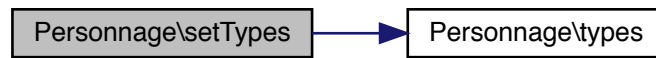


6.3.3.40 setTypes()

```
Personnage::setTypes (
    $types )
```

Définition à la ligne 271 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.41 `setVit()`

```
Personnage::setVit (
    $vit )
```

Définition à la ligne 301 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.3.3.42 `setXp()`

```
Personnage::setXp (
    $xp )
```

Définition à la ligne 231 du fichier `Personnage.php`.

Voici le graphe d'appel pour cette fonction :



6.3.3.43 setXp_max()

```
Personnage::setXp_max (
    $xp_max )
```

Définition à la ligne 238 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



6.3.3.44 soi()

```
Personnage::soi ( )
```

Définition à la ligne 196 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :

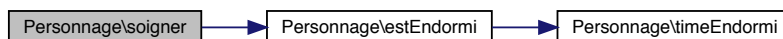


6.3.3.45 soigner()

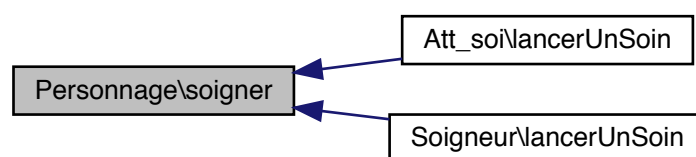
```
Personnage::soigner (
    Personnage $perso )
```

Définition à la ligne 128 du fichier Personnage.php.

Voici le graphe d'appel pour cette fonction :



Voici le graphe des appelants de cette fonction :



6.3.3.46 team()

```
Personnage::team ( )
```

Définition à la ligne 187 du fichier `Personnage.php`.

Voici le graphe des appelants de cette fonction :



6.3.3.47 ten()

`Personnage::ten ()`

Définition à la ligne 195 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :

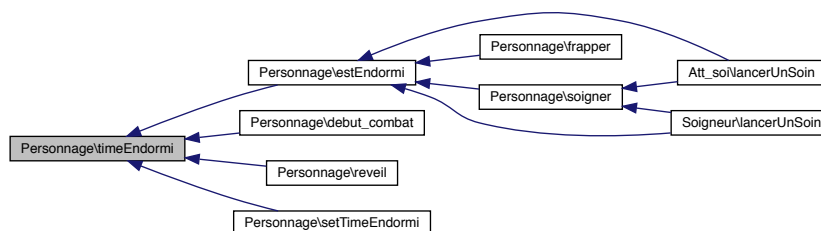


6.3.3.48 timeEndormi()

`Personnage::timeEndormi ()`

Définition à la ligne 185 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :



6.3.3.49 types()

`Personnage::types ()`

Définition à la ligne 186 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :



6.3.3.50 vit()

```
Personnage::vit ( )
```

Définition à la ligne 189 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :

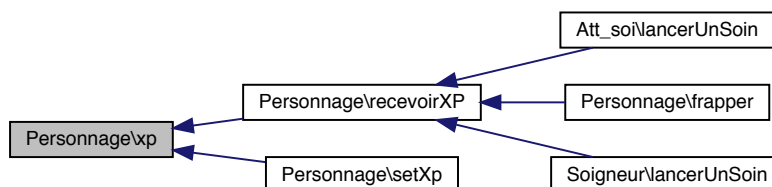


6.3.3.51 xp()

```
Personnage::xp ( )
```

Définition à la ligne 179 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :

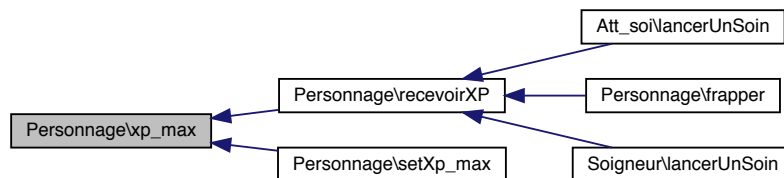


6.3.3.52 xp_max()

```
Personnage::xp_max ( )
```

Définition à la ligne 180 du fichier Personnage.php.

Voici le graphe des appelants de cette fonction :



6.3.4 Documentation des champs

6.3.4.1 \$arm

```
Personnage::$arm
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.2 \$atout

```
Personnage::$atout [protected]
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.3 \$crit

```
Personnage::$crit
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.4 \$def

```
Personnage::$def
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.5 \$esc

```
Personnage::$esc
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.6 \$etat

```
Personnage::$etat
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.7 \$etoile

```
Personnage::$etoile
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.8 \$id_perso

```
Personnage::$id_perso
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.9 \$id_user

```
Personnage::$id_user
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.10 \$niveau

`Personnage::$niveau`

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.11 \$pen

`Personnage::$pen`

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.12 \$pre

`Personnage::$pre`

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.13 \$qualite

`Personnage::$qualite`

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.14 \$soi

`Personnage::$soi`

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.15 \$team

`Personnage::$team`

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.16 \$ten

```
Personnage::$ten
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.17 \$timeEndormi

```
Personnage::$timeEndormi
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.18 \$vit

```
Personnage::$vit
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.19 \$xp

```
Personnage::$xp
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.20 \$xp_max

```
Personnage::$xp_max
```

Définition à la ligne 15 du fichier Personnage.php.

6.3.4.21 CEST_MOI

```
const Personnage::CEST_MOI = 1
```

Définition à la ligne 36 du fichier Personnage.php.

6.3.4.22 PAS_DE_MAGIE

```
const Personnage::PAS_DE_MAGIE = 5
```

Définition à la ligne 40 du fichier Personnage.php.

6.3.4.23 PERSO_ENDORMI

```
const Personnage::PERSO_ENDORMI = 6
```

Définition à la ligne 41 du fichier Personnage.php.

6.3.4.24 PERSO_FULL_LIFE

```
const Personnage::PERSO_FULL_LIFE = 8
```

Définition à la ligne 43 du fichier Personnage.php.

6.3.4.25 PERSONNAGE_ENSORCELE

```
const Personnage::PERSONNAGE_ENSORCELE = 4
```

Définition à la ligne 39 du fichier Personnage.php.

6.3.4.26 PERSONNAGE_FRAPPE

```
const Personnage::PERSONNAGE_FRAPPE = 3
```

Définition à la ligne 38 du fichier Personnage.php.

6.3.4.27 PERSONNAGE_SOIGNE

```
const Personnage::PERSONNAGE_SOIGNE = 7
```

Définition à la ligne 42 du fichier Personnage.php.

6.3.4.28 PERSONNAGE_TUE

```
const Personnage::PERSONNAGE_TUE = 2
```

Définition à la ligne 37 du fichier Personnage.php.

La documentation de cette classe a été générée à partir du fichier suivant :

— class/[Personnage.php](#)

6.4 Référence de la classe PersonnagesManager

Fonctions membres publiques

- [__construct](#) (\$db)
- [add](#) ([Base_personnage](#) \$perso, [User](#) \$user, [StuffManager](#) \$managerStuff)
- [count](#) ()
- [exists](#) (\$info)
- [get_id_team](#) (\$info, \$user)
- [get](#) (\$info)
- [get_base_perso](#) (\$id)
- [getListCheat](#) ()
- [getList](#) ([User](#) \$user)
- [getList_up](#) ([User](#) \$user, [Personnage](#) \$perso)
- [delete](#) ([Personnage](#) \$perso)
- [sup_team](#) ()
- [update_team](#) (\$team, \$id_perso)
- [update](#) ([Personnage](#) \$perso)
- [setDb](#) (\$db)

6.4.1 Description détaillée

Définition à la ligne 3 du fichier PersonnagesManager.php.

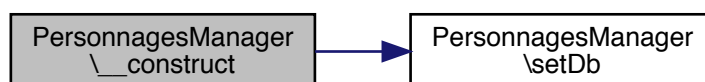
6.4.2 Documentation des constructeurs et destructeur

6.4.2.1 __construct()

```
PersonnagesManager::__construct (
    $db )
```

Définition à la ligne 7 du fichier PersonnagesManager.php.

Voici le graphe d'appel pour cette fonction :



6.4.3 Documentation des fonctions membres

6.4.3.1 add()

```
PersonnagesManager::add (
    Base_personnage $perso,
    User $user,
    StuffManager $managerStuff )
```

Définition à la ligne 12 du fichier PersonnagesManager.php.

6.4.3.2 count()

```
PersonnagesManager::count ( )
```

Définition à la ligne 56 du fichier PersonnagesManager.php.

6.4.3.3 delete()

```
PersonnagesManager::delete (
    Personnage $perso )
```

Définition à la ligne 271 du fichier PersonnagesManager.php.

6.4.3.4 exists()

```
PersonnagesManager::exists (
    $info )
```

Définition à la ligne 61 du fichier PersonnagesManager.php.

6.4.3.5 get()

```
PersonnagesManager::get (
    $info )
```

Définition à la ligne 86 du fichier PersonnagesManager.php.

6.4.3.6 get_base_perso()

```
PersonnagesManager::get_base_perso (
    $id )
```

Définition à la ligne 127 du fichier PersonnagesManager.php.

6.4.3.7 get_id_team()

```
PersonnagesManager::get_id_team (
    $info,
    $user )
```

Définition à la ligne 69 du fichier PersonnagesManager.php.

6.4.3.8 getList()

```
PersonnagesManager::getList (
    User $user )
```

Définition à la ligne 176 du fichier PersonnagesManager.php.

6.4.3.9 getList_up()

```
PersonnagesManager::getList_up (
    User $user,
    Personnage $perso )
```

Définition à la ligne 221 du fichier PersonnagesManager.php.

6.4.3.10 getListCheat()

```
PersonnagesManager::getListCheat ( )
```

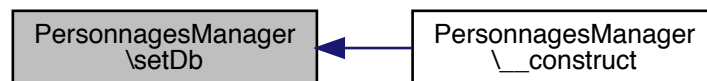
Définition à la ligne 150 du fichier PersonnagesManager.php.

6.4.3.11 setDb()

```
PersonnagesManager::setDb (
    $db )
```

Définition à la ligne 321 du fichier PersonnagesManager.php.

Voici le graphe des appelants de cette fonction :



6.4.3.12 sup_team()

```
PersonnagesManager::sup_team ( )
```

Définition à la ligne 279 du fichier PersonnagesManager.php.

6.4.3.13 update()

```
PersonnagesManager::update (
    Personnage $perso )
```

Définition à la ligne 299 du fichier PersonnagesManager.php.

6.4.3.14 update_team()

```
PersonnagesManager::update_team (
    $team,
    $id_perso )
```

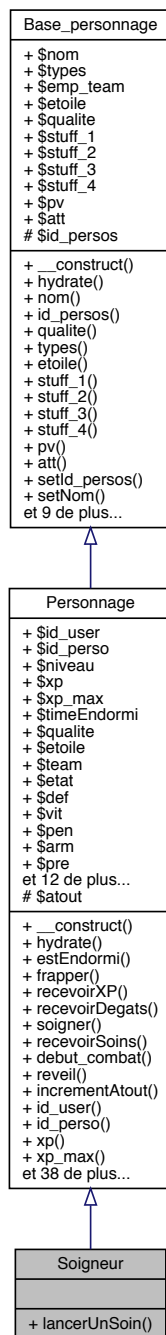
Définition à la ligne 287 du fichier PersonnagesManager.php.

La documentation de cette classe a été générée à partir du fichier suivant :

— class/[PersonnagesManager.php](#)

6.5 Référence de la classe Soigneur

Graphe d'héritage de Soigneur:



Fonctions membres publiques

— `lancerUnSoin` (`Personnage $perso`)

Membres hérités additionnels

6.5.1 Description détaillée

Définition à la ligne 3 du fichier Soigneur.php.

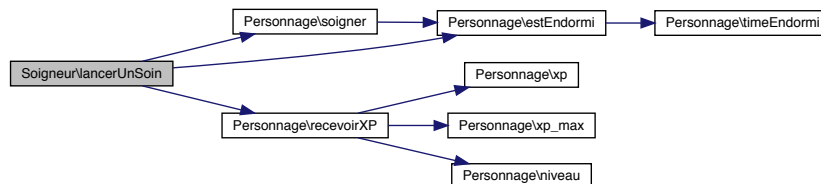
6.5.2 Documentation des fonctions membres

6.5.2.1 lancerUnSoin()

```
Soigneur::lancerUnSoin (
    Personnage $perso )
```

Définition à la ligne 5 du fichier Soigneur.php.

Voici le graphe d'appel pour cette fonction :



La documentation de cette classe a été générée à partir du fichier suivant :
 — [class/Soigneur.php](#)

6.6 Référence de la classe Stuff

Fonctions membres publiques

```

— __construct ($valeurs=array())
— hydrate (array $donnees)
— lvlUp (User $user, $tab)
— nom ()
— id_stuff ()
— id_base ()
— niveau ()
— qualite ()
— types ()
— cout ()
— etoile ()
— atr_1 ()
— atr_2 ()
— descr ()
— setId_stuff ($id_stuff)
— setId_base ($id_base)
— setNiveau ($niveau)
— setCout ($cout)
— setNom ($nom)
— setQualite ($qualite)
— setDescr ($descr)
— setTypes ($types)
— setEtoile ($etoile)
— setAtr_1 ($atr_1)
— setAtr_2 ($atr_2)
```


Champs de données

- \$nom
- \$types
- \$cout
- \$etoile
- \$niveau
- \$qualite
- \$atr_1
- \$atr_2
- \$id_base
- \$descr

Attributs protégés

- \$id_stuff

6.6.1 Description détaillée

Définition à la ligne 3 du fichier Stuff.php.

6.6.2 Documentation des constructeurs et destructeur

6.6.2.1 __construct()

```
Stuff::__construct (
    $valeurs = array() )
```

Définition à la ligne 17 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



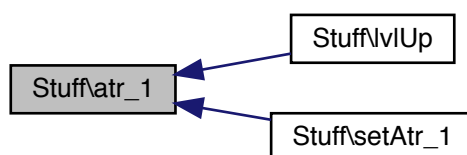
6.6.3 Documentation des fonctions membres

6.6.3.1 atr_1()

```
Stuff::atr_1 ( )
```

Définition à la ligne 66 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :

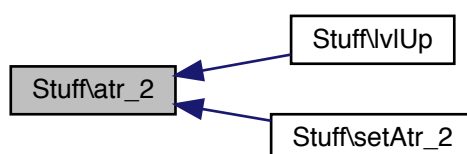


6.6.3.2 atr_2()

```
Stuff::atr_2 ( )
```

Définition à la ligne 67 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.3.3 cout()

```
Stuff::cout ( )
```

Définition à la ligne 64 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.3.4 descr()

```
Stuff::descr ( )
```

Définition à la ligne 68 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.3.5 etoile()

```
Stuff::etoile ( )
```

Définition à la ligne 65 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.3.6 hydrate()

```
Stuff::hydrate (
    array $donnees )
```

Définition à la ligne 25 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.3.7 id_base()

```
Stuff::id_base ( )
```

Définition à la ligne 60 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.3.8 id_stuff()

```
Stuff::id_stuff ( )
```

Définition à la ligne 59 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :

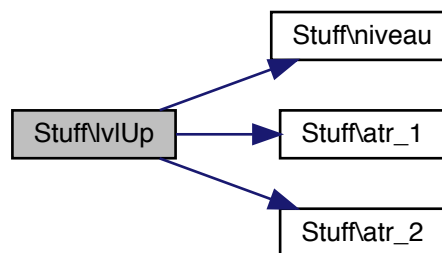


6.6.3.9 lvlUp()

```
Stuff::lvlUp (
    User $user,
    $tab )
```

Définition à la ligne 41 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :

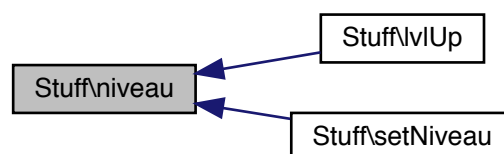


6.6.3.10 niveau()

```
Stuff::niveau ( )
```

Définition à la ligne 61 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.3.11 nom()

```
Stuff::nom ( )
```

Définition à la ligne 58 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.3.12 qualite()

```
Stuff::qualite ( )
```

Définition à la ligne 62 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.3.13 setAtr_1()

```
Stuff::setAtr_1 (
    $atr_1 )
```

Définition à la ligne 146 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



6.6.3.14 setAtr_2()

```
Stuff::setAtr_2 (
    $atr_2 )
```

Définition à la ligne 157 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



6.6.3.15 setCout()

```
Stuff::setCout (
    $cout )
```

Définition à la ligne 90 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



6.6.3.16 setDescr()

```
Stuff::setDescr (
    $descr )
```

Définition à la ligne 116 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



6.6.3.17 setEtoile()

```
Stuff::setEtoile (
    $etoile )
```

Définition à la ligne 136 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



6.6.3.18 setId_base()

```
Stuff::setId_base (
    $id_base )
```

Définition à la ligne 78 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



6.6.3.19 setId_stuff()

```
Stuff::setId_stuff (
    $id_stuff )
```

Définition à la ligne 72 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



6.6.3.20 setNiveau()

```
Stuff::setNiveau (
    $niveau )
```

Définition à la ligne 84 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



6.6.3.21 setNom()

```
Stuff::setNom (
    $nom )
```

Définition à la ligne 96 du fichier Stuff.php.

Voici le graphe d'appel pour cette fonction :



6.6.3.22 `setQualite()`

```
Stuff::setQualite (
    $qualite )
```

Définition à la ligne 106 du fichier `Stuff.php`.

Voici le graphe d'appel pour cette fonction :



6.6.3.23 `setTypes()`

```
Stuff::setTypes (
    $types )
```

Définition à la ligne 126 du fichier `Stuff.php`.

Voici le graphe d'appel pour cette fonction :



6.6.3.24 types()

```
Stuff::types ( )
```

Définition à la ligne 63 du fichier Stuff.php.

Voici le graphe des appelants de cette fonction :



6.6.4 Documentation des champs

6.6.4.1 \$atr_1

```
Stuff::$atr_1
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.2 \$atr_2

```
Stuff::$atr_2
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.3 \$cout

```
Stuff::$cout
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.4 \$descr

```
Stuff::$descr
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.5 \$etoile

```
Stuff::$etoile
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.6 \$id_base

```
Stuff::$id_base
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.7 \$id_stuff

```
Stuff::$id_stuff [protected]
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.8 \$niveau

```
Stuff::$niveau
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.9 \$nom

```
Stuff::$nom
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.10 \$qualite

```
Stuff::$qualite
```

Définition à la ligne 5 du fichier Stuff.php.

6.6.4.11 \$types

```
Stuff::$types
```

Définition à la ligne 5 du fichier Stuff.php.

La documentation de cette classe a été générée à partir du fichier suivant :

— class/[Stuff.php](#)

6.7 Référence de la classe StuffManager

Fonctions membres publiques

- [__construct](#) (\$db)
- [add](#) (Stuff \$stuff)
- [count](#) ()
- [exists](#) (\$info)
- [get_niveau_sup](#) (\$info)
- [get](#) (\$info)
- [get_base_perso](#) (\$id)
- [getListCheat](#) ()
- [getList](#) (User \$user)
- [delete](#) (Personnage \$perso)
- [sup_team](#) ()
- [update_team](#) (\$team, \$id_perso)
- [update](#) (Stuff \$stuff)
- [setDb](#) (\$db)

6.7.1 Description détaillée

Définition à la ligne 3 du fichier StuffManager.php.

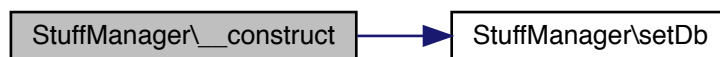
6.7.2 Documentation des constructeurs et destructeur

6.7.2.1 __construct()

```
StuffManager::__construct (
    $db )
```

Définition à la ligne 7 du fichier StuffManager.php.

Voici le graphe d'appel pour cette fonction :



6.7.3 Documentation des fonctions membres

6.7.3.1 add()

```
StuffManager::add (
    Stuff $stuff )
```

Définition à la ligne 12 du fichier StuffManager.php.

6.7.3.2 count()

```
StuffManager::count ( )
```

Définition à la ligne 45 du fichier StuffManager.php.

6.7.3.3 delete()

```
StuffManager::delete (
    Personnage $perso )
```

Définition à la ligne 189 du fichier StuffManager.php.

6.7.3.4 exists()

```
StuffManager::exists (
    $info )
```

Définition à la ligne 50 du fichier StuffManager.php.

6.7.3.5 get()

```
StuffManager::get (
    $info )
```

Définition à la ligne 72 du fichier StuffManager.php.

6.7.3.6 get_base_perso()

```
StuffManager::get_base_perso (
    $id )
```

Définition à la ligne 95 du fichier StuffManager.php.

6.7.3.7 get_niveau_sup()

```
StuffManager::get_niveau_sup (
    $info )
```

Définition à la ligne 58 du fichier StuffManager.php.

6.7.3.8 getList()

```
StuffManager::getList (
    User $user )
```

Définition à la ligne 144 du fichier StuffManager.php.

6.7.3.9 getListCheat()

```
StuffManager::getListCheat ( )
```

Définition à la ligne 118 du fichier StuffManager.php.

6.7.3.10 setDb()

```
StuffManager::setDb (
    $db )
```

Définition à la ligne 229 du fichier StuffManager.php.

Voici le graphe des appelants de cette fonction :



6.7.3.11 sup_team()

```
StuffManager::sup_team ( )
```

Définition à la ligne 194 du fichier StuffManager.php.

6.7.3.12 update()

```
StuffManager::update (
    Stuff $stuff )
```

Définition à la ligne 214 du fichier StuffManager.php.

6.7.3.13 update_team()

```
StuffManager::update_team (
    $team,
    $id_perso )
```

Définition à la ligne 202 du fichier StuffManager.php.

La documentation de cette classe a été générée à partir du fichier suivant :

— class/[StuffManager.php](#)

6.8 Référence de la classe User

Fonctions membres publiques

- `__construct` (\$valeurs=array())
- `hydrate` (array \$donnees)
- `payer_argent` (\$somme)
- `id_user` ()
- `pseudo` ()
- `endu` ()
- `endu_Max` ()
- `mdp` ()
- `niveau` ()
- `email` ()
- `xp` ()
- `xp_Max` ()
- `or_` ()
- `argent` ()
- `emp_team_max` ()
- `nomValide` ()
- `setId_user` (\$id_user)
- `setPseudo` (\$pseudo)
- `setMdp` (\$mdp)
- `setNiveau` (\$niveau)
- `setXp` (\$xp)
- `setXp_Max` (\$xp_Max)
- `setEndu` (\$endu)
- `setEndu_Max` (\$endu_Max)
- `setOr_` (\$or_)
- `setArgent` (\$argent)
- `setEmail` (\$email)
- `setEmp_team_max` (\$emp_team_max)

Champs de données

- `$pseudo`
- `$email`
- `$mdp`
- `$niveau`
- `$or_`
- `$argent`
- `$xp`
- `$Xp_Max`
- `$endu`
- `$endu_Max`
- `$emp_team_max`

Attributs protégés

- `$id_user`

6.8.1 Description détaillée

Définition à la ligne 3 du fichier User.php.

6.8.2 Documentation des constructeurs et destructeur

6.8.2.1 __construct()

```
User::__construct (
    $valeurs = array() )
```

Définition à la ligne 19 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



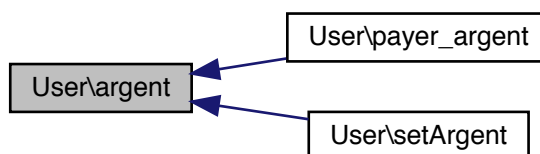
6.8.3 Documentation des fonctions membres

6.8.3.1 argent()

```
User::argent ( )
```

Définition à la ligne 60 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.2 email()

```
User::email ( )
```

Définition à la ligne 56 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.3 emp_team_max()

```
User::emp_team_max ( )
```

Définition à la ligne 61 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.4 endu()

```
User::endu ( )
```

Définition à la ligne 52 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.5 `endu_Max()`

```
User::endu_Max ( )
```

Définition à la ligne 53 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.6 `hydrate()`

```
User::hydrate (
    array $donnees )
```

Définition à la ligne 27 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.7 `id_user()`

```
User::id_user ( )
```

Définition à la ligne 50 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.8 mdp()

```
User::mdp ( )
```

Définition à la ligne 54 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.9 niveau()

```
User::niveau ( )
```

Définition à la ligne 55 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.10 nomValide()

```
User::nomValide ( )
```

Définition à la ligne 63 du fichier User.php.

6.8.3.11 or_()

```
User::or_ ( )
```

Définition à la ligne 59 du fichier User.php.

Voici le graphe des appelants de cette fonction :

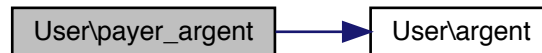


6.8.3.12 payer_argent()

```
User::payer_argent (
    $somme )
```

Définition à la ligne 43 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.13 pseudo()

```
User::pseudo ( )
```

Définition à la ligne 51 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.14 setArgent()

```
User::setArgent (
    $argent )
```

Définition à la ligne 157 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.15 setEmail()

```
User::setEmail (
    $email )
```

Définition à la ligne 168 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.16 setEmp_team_max()

```
User::setEmp_team_max (
    $emp_team_max )
```

Définition à la ligne 178 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.17 `setEndu()`

```
User::setEndu (
    $endu )
```

Définition à la ligne 127 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.18 `setEndu_Max()`

```
User::setEndu_Max (
    $endu_Max )
```

Définition à la ligne 135 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.19 setId_user()

```
User::setId_user (
    $id_user )
```

Définition à la ligne 68 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.20 setMdp()

```
User::setMdp (
    $mdp )
```

Définition à la ligne 84 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.21 setNiveau()

```
User::setNiveau (
    $niveau )
```

Définition à la ligne 94 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.22 `setOr_()`

```
User::setOr_ (  
    $or_ )
```

Définition à la ligne 146 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.23 `setPseudo()`

```
User::setPseudo (  
    $pseudo )
```

Définition à la ligne 74 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.24 setXp()

```
User::setXp (
    $xp )
```

Définition à la ligne 105 du fichier User.php.

Voici le graphe d'appel pour cette fonction :



6.8.3.25 setXp_Max()

```
User::setXp_Max (
    $xp_Max )
```

Définition à la ligne 116 du fichier User.php.

6.8.3.26 xp()

```
User::xp ( )
```

Définition à la ligne 57 du fichier User.php.

Voici le graphe des appelants de cette fonction :



6.8.3.27 xp_Max()

```
User::xp_Max ( )
```

Définition à la ligne 58 du fichier User.php.

6.8.4 Documentation des champs

6.8.4.1 \$argent

```
User::$argent
```

Définition à la ligne 5 du fichier User.php.

6.8.4.2 \$email

```
User::$email
```

Définition à la ligne 5 du fichier User.php.

6.8.4.3 \$emp_team_max

```
User::$emp_team_max
```

Définition à la ligne 5 du fichier User.php.

6.8.4.4 \$endu

```
User::$endu
```

Définition à la ligne 5 du fichier User.php.

6.8.4.5 \$endu_Max

```
User::$endu_Max
```

Définition à la ligne 5 du fichier User.php.

6.8.4.6 \$id_user

```
User::$id_user [protected]
```

Définition à la ligne 5 du fichier User.php.

6.8.4.7 \$mdp

```
User::$mdp
```

Définition à la ligne 5 du fichier User.php.

6.8.4.8 \$niveau

```
User::$niveau
```

Définition à la ligne 5 du fichier User.php.

6.8.4.9 \$or_

```
User::$or_
```

Définition à la ligne 5 du fichier User.php.

6.8.4.10 \$pseudo

```
User::$pseudo
```

Définition à la ligne 5 du fichier User.php.

6.8.4.11 \$xp

```
User::$xp
```

Définition à la ligne 5 du fichier User.php.

6.8.4.12 \$Xp_Max

```
User::$Xp_Max
```

Définition à la ligne 5 du fichier User.php.

La documentation de cette classe a été générée à partir du fichier suivant :

— class/[User.php](#)

6.9 Référence de la classe UserManager

Fonctions membres publiques

- [__construct](#) (\$db)
- [add](#) ([User](#) \$user)
- [verifMdp](#) (\$pseudo, \$pass)
- [count](#) ()
- [delete](#) ([User](#) \$user)
- [exists](#) (\$info)
- [get](#) (\$info)
- [getList](#) (\$pseudo)
- [update](#) ([User](#) \$user)
- [setDb](#) (\$db)

6.9.1 Description détaillée

Définition à la ligne 3 du fichier UserManager.php.

6.9.2 Documentation des constructeurs et destructeur

6.9.2.1 __construct()

```
UserManager::__construct (
    $db )
```

Définition à la ligne 7 du fichier UserManager.php.

Voici le graphe d'appel pour cette fonction :



6.9.3 Documentation des fonctions membres

6.9.3.1 add()

```
UserManager::add (
    User $user )
```

Définition à la ligne 12 du fichier UserManager.php.

6.9.3.2 count()

```
UserManager::count ( )
```

Définition à la ligne 57 du fichier UserManager.php.

6.9.3.3 delete()

```
UserManager::delete (
    User $user )
```

Définition à la ligne 62 du fichier UserManager.php.

6.9.3.4 exists()

```
UserManager::exists (
    $info )
```

Définition à la ligne 67 du fichier UserManager.php.

6.9.3.5 get()

```
UserManager::get (
    $info )
```

Définition à la ligne 82 du fichier UserManager.php.

6.9.3.6 getList()

```
UserManager::getList (
    $pseudo )
```

Définition à la ligne 110 du fichier UserManager.php.

6.9.3.7 setDb()

```
UserManager::setDb (
    $db )
```

Définition à la ligne 143 du fichier UserManager.php.

Voici le graphe des appelants de cette fonction :



6.9.3.8 update()

```
UserManager::update (
    User $user )
```

Définition à la ligne 126 du fichier UserManager.php.

6.9.3.9 verifMdp()

```
UserManager::verifMdp (
    $pseudo,
    $pass )
```

Définition à la ligne 43 du fichier UserManager.php.

La documentation de cette classe a été générée à partir du fichier suivant :

— class/[UserManager.php](#)

Chapitre 7

Documentation des fichiers

7.1 Référence du fichier ajax/add_perso.php

Fonctions

— `chargerClasse` (\$classe)

Variables

— `$manager` = new `PersonnagesManager`(\$db)
— `$usermanager` = new `UserManager`(\$db)
— `$managerStuff` = new `StuffManager`(\$db)
— `$perso` = \$manager->get_base_perso((int) \$_POST['id_perso'])
— `$user` = \$usermanager->get((int) \$_POST['id_user'])

7.1.1 Documentation des fonctions

7.1.1.1 `chargerClasse()`

```
chargerClasse (
    $classe )
```

Définition à la ligne 3 du fichier add_perso.php.

7.1.2 Documentation des variables

7.1.2.1 \$manager

```
$manager = new PersonnagesManager($db)
```

Définition à la ligne 13 du fichier add_perso.php.

7.1.2.2 \$managerStuff

```
$managerStuff = new StuffManager($db)
```

Définition à la ligne 15 du fichier add_perso.php.

7.1.2.3 \$perso

```
$perso = $manager->get_base_perso((int) $_POST['id_perso'])
```

Définition à la ligne 17 du fichier add_perso.php.

7.1.2.4 \$user

```
$user = $usermanager->get((int) $_POST['id_user'])
```

Définition à la ligne 19 du fichier add_perso.php.

7.1.2.5 \$usermanager

```
$usermanager = new UserManager($db)
```

Définition à la ligne 14 du fichier add_perso.php.

7.2 Référence du fichier ajax/gestion_perso.php

Variables

— `if(isset($_SESSION['tab_perso'])) if(isset($_POST["tab"])) else`

7.2.1 Documentation des variables

7.2.1.1 else

```
if (isset( $_SESSION[ 'tab_perso' ] )) if (isset( $_POST[ "tab" ] )) else
```

Valeur initiale :

```
{
    $_SESSION[ 'tab_perso' ] = []
```

Définition à la ligne 9 du fichier gestion_perso.php.

7.3 Référence du fichier ajax/gestion_stuff.php

Fonctions

— [chargerClasse](#) (\$classe)

Variables

```
— $manager = new PersonnagesManager($db)
— $managerUser = new UserManager($db)
— $managerStuff = new StuffManager($db)
— $stuff = $managerStuff->get($_POST["id_stuff"])
— $tab = $managerStuff->get_niveau_sup($stuff->niveau())
— $perso = $manager->get($_POST["id_perso"])
— if($stuff->cout()<=$_SESSION['user']->argent()) else
```

7.3.1 Documentation des fonctions

7.3.1.1 chargerClasse()

```
chargerClasse (
    $classe )
```

Définition à la ligne 2 du fichier gestion_stuff.php.

7.3.2 Documentation des variables

7.3.2.1 \$manager

```
$manager = new PersonnagesManager($db)
```

Définition à la ligne 13 du fichier gestion_stuff.php.

7.3.2.2 \$managerStuff

```
$managerStuff = new StuffManager($db)
```

Définition à la ligne 15 du fichier gestion_stuff.php.

7.3.2.3 \$managerUser

```
$managerUser = new UserManager($db)
```

Définition à la ligne 14 du fichier gestion_stuff.php.

7.3.2.4 \$perso

```
$perso = $manager->get($_POST["id_perso"])
```

Définition à la ligne 18 du fichier gestion_stuff.php.

7.3.2.5 \$stuff

```
$stuff = $managerStuff->get($_POST["id_stuff"])
```

Définition à la ligne 16 du fichier gestion_stuff.php.

7.3.2.6 \$tab

```
$tab = $managerStuff->get_niveau_sup($stuff->niveau())
```

Définition à la ligne 17 du fichier gestion_stuff.php.

7.3.2.7 else

```
if ( $stuff->cout() <= $_SESSION[ 'user' ]->argent() ) else
```

Valeur initiale :

```
{  
    echo "Pas assez d'argent"
```

Définition à la ligne 56 du fichier gestion_stuff.php.

7.4 Référence du fichier ajax/gestion_sup_tab.php

7.5 Référence du fichier ajax/gestion_up_perso.php

Fonctions

— [chargerClasse](#) (\$classe)

Variables

— [\\$manager](#) = new [PersonnagesManager](#)(\$db)
— [\\$perso](#) = \$manager->get(\$_POST["btn_valid"])
— [\\$entry](#)
— [\\$json](#)[0] = \$entry

7.5.1 Documentation des fonctions

7.5.1.1 chargerClasse()

```
chargerClasse (  
    $classe )
```

Définition à la ligne 2 du fichier gestion_up_perso.php.

7.5.2 Documentation des variables

7.5.2.1 \$entry

\$entry

Valeur initiale :

```
= array(
    'result' => 'lvl Up',
    'niveau' => utf8_encode($perso->niveau()),
    'id_perso' => utf8_encode($perso->id_perso()),
    'type' => utf8_encode($perso->types()),
    'perso_pv' => utf8_encode($perso->pv()),
    'perso_att' => utf8_encode($perso->att()),
    'perso_def' => utf8_encode($perso->def()),
    'perso_vit' => utf8_encode($perso->vit()),
    'perso_pen' => utf8_encode($perso->pen()),
    'perso_arm' => utf8_encode($perso->arm()),
    'perso_pre' => utf8_encode($perso->pre()),
    'perso_esc' => utf8_encode($perso->esc()),
    'perso_crit' => utf8_encode($perso->crit()),
    'perso_ten' => utf8_encode($perso->ten()),
    'perso_soi' => utf8_encode($perso->soi()),
    'tab' => $_SESSION['tab_perso']
)
```

Définition à la ligne 28 du fichier gestion_up_perso.php.

7.5.2.2 \$json

\$json[0] = \$entry

Définition à la ligne 47 du fichier gestion_up_perso.php.

7.5.2.3 \$manager

\$manager = new [PersonnagesManager](#)(\$db)

Définition à la ligne 13 du fichier gestion_up_perso.php.

7.5.2.4 \$perso

\$perso = \$manager->get(\$_POST["btn_valid"])

Définition à la ligne 14 du fichier gestion_up_perso.php.

7.6 Référence du fichier ajax/sup_perso.php

Fonctions

— [chargerClasse](#) (\$classe)

Variables

- `$manager` = new `PersonnagesManager`(\$db)
- `$perso` = `$manager->get((int) $_POST['id_perso'])`

7.6.1 Documentation des fonctions

7.6.1.1 `chargerClasse()`

```
chargerClasse (  
    $classe )
```

Définition à la ligne 3 du fichier sup_perso.php.

7.6.2 Documentation des variables

7.6.2.1 `$manager`

```
$manager = new PersonnagesManager ($db)
```

Définition à la ligne 13 du fichier sup_perso.php.

7.6.2.2 `$perso`

```
$perso = $manager->get((int) $_POST['id_perso'])
```

Définition à la ligne 15 du fichier sup_perso.php.

7.7 Référence du fichier ajax/team_ajax.php

Fonctions

- `chargerClasse` (\$classe)

Variables

- `$manager` = new `PersonnagesManager`(\$db)

7.7.1 Documentation des fonctions

7.7.1.1 chargerClasse()

```
chargerClasse (
    $classe )
```

Définition à la ligne 2 du fichier team_ajax.php.

7.7.2 Documentation des variables

7.7.2.1 \$manager

```
$manager = new PersonnagesManager ($db)
```

Définition à la ligne 12 du fichier team_ajax.php.

7.8 Référence du fichier class/Att_soi.php

Structures de données

— class [Att_soi](#)

7.9 Référence du fichier class/Base_personnage.php

Class de base des personnages.

Structures de données

— class [Base_personnage](#)

7.9.1 Description détaillée

Class de base des personnages.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Class de base des personnages. Elle définit les attributs et methodes inhérent à tous les personnages.

7.10 Référence du fichier class/Personnage.php

Class des personnages.

Structures de données

— class [Personnage](#)

7.10.1 Description détaillée

Class des personnages.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Class des personnages. Elle définit les attributs et methodes des personnages.

7.11 Référence du fichier class/PersonnagesManager.php

Structures de données

— class [PersonnagesManager](#)

7.12 Référence du fichier class/Soigneur.php

Structures de données

— class [Soigneur](#)

7.13 Référence du fichier class/Stuff.php

Structures de données

— class [Stuff](#)

7.14 Référence du fichier includes/stuff.php

Menu de gestion du stuff.

7.14.1 Description détaillée

Menu de gestion du stuff.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Ce fichier est le menu de gestion du stuff, il permet de gérer toutes les fonctionnalités concernant le stuff.

7.15 Référence du fichier class/StuffManager.php

Structures de données

— class [StuffManager](#)

7.16 Référence du fichier class/User.php

Structures de données

— class [User](#)

7.17 Référence du fichier class/UserManager.php

Structures de données

— class [UserManager](#)

7.18 Référence du fichier includes/cheat.php

Menu cheat du jeu.

7.18.1 Description détaillée

Menu cheat du jeu.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Ce fichier est le menu cheat du jeu, ce menu permet de cheater afin de tester toutes les fonctionnalités du jeu.

7.19 Référence du fichier includes/cheat/modif_stuff.php

Menu cheat du stuff.

7.19.1 Description détaillée

Menu cheat du stuff.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

A faire Finir l'interface

Ce fichier est le menu cheat du stuff, ce menu permet de cheater afin de tester toutes les fonctionnalités concernant le stuff.

7.20 Référence du fichier includes/cheat/modif_team.php

Menu cheat de la team.

7.20.1 Description détaillée

Menu cheat de la team.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Ce fichier est le menu cheat de la team, ce menu permet de cheater afin de tester toutes les fonctionnalités concernant votre équipe.

7.21 Référence du fichier includes/eviterMessageAvertissement.php

7.22 Référence du fichier includes/Function_Attaque.php

Définit les fonctions d'un combat.

7.22.1 Description détaillée

Définit les fonctions d'un combat.

Auteur

Guillaume Nagiel

Version

1.0

Date

1 Mars 2018

Ce fichier définit toutes les fonctions utilisées pour effectuer un combat.

7.23 Référence du fichier includes/Function_Jeu.php

Définit les fonctions du jeu.

Fonctions

- `affichePerso` (\$perso)
- `fonctionComparaison` (\$a, \$b)

7.23.1 Description détaillée

Définit les fonctions du jeu.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Ce fichier définit toutes les fonctions utilisées pour initialiser le jeu.

7.23.2 Documentation des fonctions

7.23.2.1 `affichePerso()`

```
affichePerso (
    $perso )
```

Afficher un personnage

Génère l'HTML permettant d'afficher un personnage du jeu. (cf `$perso`)

Paramètres

<code>\$perso</code>	Perso à afficher.
----------------------	-------------------

Renvoie

Un block HTML.

Définition à la ligne 13 du fichier Function_Jeu.php.

7.23.2.2 fonctionComparaison()

```
fonctionComparaison (
    $a,
    $b )
```

Définition à la ligne 43 du fichier Function_Jeu.php.

7.24 Référence du fichier includes/identifiant.php

Connexion PDO.

7.24.1 Description détaillée

Connexion PDO.

Auteur

Guillaume Nagiel

Version

1.0

Date

29 Janvier 2018

Ce fichier contient le code permettant d'instancier l'objet PDO afin de se connecter à la base de données.

7.25 Référence du fichier includes/stuff_modal.php

Modal de gestion d'un équipement.

7.25.1 Description détaillée

Modal de gestion d'un équipement.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

A faire Implémenter la fonctionnalité pour up plusieurs niveaux

Ce fichier est un modal gérant les fonctionnalités d'un équipement en particulier. Il est appelé par le fichier stuff.php.

7.26 Référence du fichier includes/team - save.php

7.27 Référence du fichier includes/team.php

Menu de gestion d'équipe.

Fonctions

- `cadre_team` (\$emp_team, \$user, \$manager)
Génération de l'équipe.

7.27.1 Description détaillée

Menu de gestion d'équipe.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Ce menu permet de gérer son équipe.

7.27.2 Documentation des fonctions

7.27.2.1 `cadre_team()`

```
cadre_team (
    $emp_team,
    $user,
    $manager )
```

Génération de l'équipe.

Permet générer l'équipe existante, en implémentant les fonction de drag and drop.

Paramètres

<code>\$emp_team</code>	Emplacement du personnage (de 1 à 6).
<code>\$user</code>	Class User .
<code>\$manager</code>	Class PersonnagesManager .

Renvoie

un block html pour chaque personnage, définit ou non.

Définition à la ligne 85 du fichier team.php.

7.28 Référence du fichier includes/up_perso.php

Menu d'amélioration des personnages.

7.28.1 Description détaillée

Menu d'amélioration des personnages.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Ce menu permet d'améliorer ses personnages, en sacrifiant des persos inutiles.

7.29 Référence du fichier includes/up_perso_modal.php

Modal permettant la selection des personnages à sacrifier pour en up un autre.

7.29.1 Description détaillée

Modal permettant la selection des personnages à sacrifier pour en up un autre.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Ce modal permet la selection des personnages à sacrifier pour en up un autre. Il est appelé par le fichier [up_perso.php](#).

7.30 Référence du fichier index.php

Page d'index du jeu.

Fonctions

- `chargerClass` (\$class)
Charger les class à la demande.

7.30.1 Description détaillée

Page d'index du jeu.

Auteur

Guillaume Nagiel

Version

1.0

Date

26 Janvier 2018

Ce fichier est la base du jeu.

A faire Il reste à implémenter les scripts de combats PVE et PVP.

7.30.2 Documentation des fonctions

7.30.2.1 `chargerClass()`

```
chargerClass (  
    $class )
```

Charger les class à la demande.

Permet de charger les class uniquement si elles sont utilisées via la fonction `spl_autoload_register('chargerClass')`.
(cf #`$classe`)

Paramètres

<code>\$class</code>	Class à instancier.
----------------------	---------------------

Renvoie

Rien.

Définition à la ligne 28 du fichier index.php.

7.31 Référence du fichier JS/add_perso.js

Ajax pour l'ajout' de personnage.

Fonctions

- `function auto_load ()`
Chargement automatique.
- `add_perso submit (function() { s=$(this).serialize();$.ajax({ type:"POST", data:s, url:'ajax/add_perso.php', success:function(retour){ auto_load();} });return false;})`

7.31.1 Description détaillée

Ajax pour l'ajout' de personnage.

Auteur

Guillaume Nagiel

Version

1.0

Date

5 Mars 2018

Ce fichier js sert à ajouter un personnage dans son deck via une requete Ajax.

7.31.2 Documentation des fonctions

7.31.2.1 auto_load()

```
auto_load ( )
```

Chargement automatique.

Permet de recharger automatiquement la page de gestion de l'équipe.

Définition à la ligne 15 du fichier add_perso.js.

7.31.2.2 submit()

```
add_perso submit (
    function() { s=$(this).serialize();$.ajax({ type:"POST", data:s, url:'ajax/add_
perso.php', success:function(retour){ auto_load();} });return false;} )
```

7.32 Référence du fichier JS/highlight/highlight.pack.js

Variables

```
— var hljs=new function(){var q={};var a={};function o(c){return c.replace(/&/gm,"&amp;").replace(/</gm,"&lt;").replace(/>/gm,"&gt;")
l(t,s){if(!t){return false}for(var c=0;c<t.length;c++){if(t[c]==s){return true}}return false}function e(t,s,c){var
u="m"+(t.c?"i":"")+(c?"g":"");return new RegExp(s,u)}function k(s){for(var c=0;c<s.childNodes.length;c++){node=s.
childNodes[c];if(node.nodeName=="CODE"){return node}if(!(node.nodeType==3&&node.nodeValue.
match(/\s+/))){return null}}}function i(t){var s="";for(var c=0;c<t.childNodes.length;c++){if(t.childNodes[c].node
Type==3){s+=t.childNodes[c].nodeValue}else{if(t.childNodes[c].nodeName=="BR"){s+="\n"}else{s+=i(t.
childNodes[c])}}return s}function b(u){var s=u.className.split(/\s+/);s=s.concat(u.parentNode.class
Name.split(/\s+/));for(var c=0;c<s.length;c++){var t=s[c].replace(/^\s*language-/,"");if(t=="no-highlight"){throw"No
highlight"}if(q[t]){return t}}function d(c){var s=[];(function(u,v){for(var t=0;t<u.childNodes.length;t++){if(u.
childNodes[t].nodeType==3){v+=u.childNodes[t].nodeValue.length}else{if(u.childNodes[t].nodeName=="BR"){v+=1}else{s.
push({event:"start",offset:v,node:u.childNodes[t]};v=arguments.callee(u.childNodes[t],v);s.push({event:
:"stop",offset:v,node:u.childNodes[t]})}}return v})(c,0);return s}function n(A,B,z){var t=0;var y="";var
v=[];function w(){if(A.length&&B.length){if(A[0].offset!=B[0].offset){return(A[0].offset<B[0].offset)?A:
:B}else{return(A[0].event=="start"&&B[0].event=="stop"?B:A)}}else{return A.length?A:B}}function u(F){var
G="<" + F.nodeName.toLowerCase();for(var D=0;D<F.attributes.length;D++){var E=F.attributes[D];G+="
" + E.nodeName.toLowerCase();if(E.nodeValue!=undefined){G+="="+o(E.nodeValue)+""}return G+">"}function
C(D){return "<" + D.nodeName.toLowerCase() + ">"}while(A.length||B.length){var x=w().splice(0,1)[0];y+=o(z.
substr(t,x.offset-t));t=x.offset;if(x.event=="start"){y+=u(x.node);v.push(x.node)}else{if(x.event=="stop"){var
s=v.length;do{s--;var c=v[s];y+=C(c)}while(c!=x.node);v.splice(s,1);while(s<v.length){y+=u(v[s]);s++}}y+=z.
substr(t);return y}function h(K,E){function L(Q,P){Q.sm=[];for(var O=0;O<Q.c.length;O++){for(var N=0;N<P.
m.length;N++){if(P.m[N].cN==Q.c[O]){Q.sm[Q.sm.length]=P.m[N]}}}function A(N,P){if(!P.c){return null}if(!P.
sm){L(P,l)}for(var O=0;O<P.sm.length;O++){if(P.sm[O].bR.test(N)){return P.sm[O]}return null}function
x(O,N){if(D[O].e&&D[O].eR.test(N)){return 1}if(D[O].eW){var P=x(O-1,N);return P?P+1:0}return 0}function
y(N,O){return O.iR&&O.iR.test(N)}function B(T,R){var P=[];function S(U){if(!P(U)){P[P.length]=U}}if(T.
c){for(var O=0;O<R.m.length;O++){if(!T.c.R.m[O].cN){S(R.m[O].b)}var N=D.length-1;do{if(D[N].e){S(D[N].e)}N-
}while(D[N+1].eW);if(T.i){S(T.i)}var Q="("+P[0];for(var O=0;O<P.length;O++){Q+="|"+P[O]}Q+=")";return
e(R,Q)}function t(P,O){var Q=D[D.length-1];if(!Q.t){Q.t=B(Q,l)}P=P.substr(O);var N=Q.t.exec(P);if(!N){return[P,"",true]}if(N.
index==0){return["",N[0],false]}else{return[P.substr(0,N.index),N[0],false]}function s(R,N){var O=l.
cl?N[0].toLowerCase():N[0];for(var Q in R.keywordGroups){if(!R.keywordGroups.hasOwnProperty(Q)){continue}var
P=R.keywordGroups[Q].hasOwnProperty(O);if(P){return[Q,P]}return false}function G(Q,T){if(!T.k||!T.
l){return o(Q)}if(!T.lR){var S="("+T.l[0];for(var P=1;P<T.l.length;P++){S+="|"+T.l[P]}S+=")";T.lR=e(l,S,true)}var
R="";var U=0;T.lR.lastIndex=0;var O=T.lR.exec(Q);while(O){R+=o(Q.substr(U,O.index-U));var N=s(T,O);if(N){u+=N[1];R+=
'<span class="'+N[0]+'>'+o(O[0])+'</span>'}else{R+=o(O[0])}U=T.lR.lastIndex;O=T.lR.exec(Q)}R+=o(Q.
substr(U,Q.length-U));return R}function M(N,P){if(P.subLanguage&&a[P.subLanguage]){var O=h(P.sub
Language,N);u+=O.keyword_count;C+=O.r;return O.value}else{return G(N,P)}}function J(P,N){var O=P.
nM?"<span class='"+P.displayClassName+"'>":if(P.R){C+=O;P.buffer=""}}else{if(P.eB){C+=o(N)+O;P.
buffer=""}}else{C+=O;P.buffer=N}}D[D.length]=P}function F(S,O,T){var U=D[D.length-1];if(T){C+=M(U.
buffer+S,U);return false}var P=A(O,U);if(P){C+=M(U.buffer+S,U);J(P,O);C+=P.r;return P.r}var N=x(D.
length-1,O);if(N){var R=U.nM?"<span>":if(U.e){C+=M(U.buffer+S,U)+R}}else{if(U.e){C+=M(U.
buffer+S,U)+R+o(O)}else{C+=M(U.buffer+S+O,U)+R}}while(N>1){R=D[D.length-2].nM?"<span>";C+=R;N-
;D[D.length--].D.length--;D[D.length-1].buffer="";if(U.starts){for(var Q=0;Q<l.m.length;Q++){if(l.m[Q].cN==U.
starts){J(l.m[Q],"");break}}return U.r}if(y(O,U)){throw"illegal"}var l=q[K];var D=[l.dM];var C=0;var u=0;var
c="";try{var w=0;l.dM.buffer="";do{var z=t(E,w);var v=F(z[0],z[1],z[2]);w+=z[0].length;if(!v){w+=z[1].length}}while(!z[2]);if(D.
length>1){throw"illegal"}return{r:C,keyword_count:u,value:c}}catch(H){if(H=="illegal"){return{r:0,keyword_
count:0,value:o(E)}}else{throw H}}}function j(){for(var s in q){if(!q.hasOwnProperty(s)){continue}var
```

```

t=q[s];for(var c=0;c<t.m.length;c++){var u=t.m[c];if(u.b){u.bR=e(t,"^"+u.b)}if(u.e){u.eR=e(t,"^"+u.e)}if(u.↵
i){u.iR=e(t,"^(?:"+u.i+"")")t.dM.iR=e(t,"^(?:"+t.dM.i+"")");if(u.r==undefined){u.r=1}if(!u.displayClassName){u.↵
displayClassName=u.cN}}function g(){function t(w){if(!w.keywordGroups){for(var v in w.k){if(!w.k.hasOwn↵
Property(v)){continue}if(w.k[v] instanceof Object){w.keywordGroups=w.k}else{w.keywordGroups={keyword↵
:w.k}}break}}for(var s in q){if(!q.hasOwnProperty(s)){continue}var u=q[s];t(u.dM);for(var c=0;c<u.↵
m.length;c++){t(u.m[c])}}function f(){if(f.called){return f.called=true;j();g();a=q}function r(y,C){try{var ↵
F=i(y);var v=b(y)}catch(z){if(z=="No highlight"){return}}if(v){var B=h(v,F).value}else{var D=0;for(var E ↵
in a){if(!a.hasOwnProperty(E)){continue}var t=h(E,F);var c=t.keyword_count+t.r;if(c>D){D=c;var B=t.↵
value;v=E}}if(B){var x=y.className;if(!x.match(v)){x+=" "+v}var s=d(y);if(s.length){var u=document.create↵
Element("pre");u.innerHTML=B;B=n(s,d(u),F)}if(C){B=B.replace(/^((<[^\>]+>|t)+)/gm,function(G,J,I,H){return ↵
J.replace(/t/g,C)}}var A=document.createElement("div");A.innerHTML='<pre><code class="'+x+'>'+B+'</code></pre>'; ↵
w=y.parentNode.parentNode;w.replaceChild(A.firstChild,y.parentNode)}function m(){if(m.called){return m.↵
called=true;f();if(arguments.length){for(var c=0;c<arguments.length;c++){if(q[arguments[c]]){a[arguments[c]]=q[arguments[c]] ↵
t=document.getElementsByTagName("pre");for(var c=0;c<t.length;c++){var s=k(t[c]);if(s){r(s,hjs.tab↵
Replace)}}}function p(){var c=arguments;var s=function(){m.apply(null,c)};if(window.addEventListener){window.↵
addEventListener("DOMContentLoaded",s,false);window.addEventListener("load",s,false)}else{if(window.↵
attachEvent){window.attachEvent("onload",s)}else{window.onload=s}}this.LANGUAGES=q,this.initHighlightingOnLoad=p,this.↵
highlightBlock=r,this.initHighlighting=m,this.IR="[a-zA-Z][a-zA-Z0-9_]*";this.UIR="[a-zA-Z_][a-zA-Z0-9_↵
]*";this.NR="\\b\\d+(\\.\\d+)?";this.CNR="\\b(0x[A-Za-z0-9_]+|\\d+(\\.\\d+)?";this.RSR="!|=|%=|&|&&|\\*|\\*=|\\+|\\+=|, ↵
|-|=|/|=|:|<|<<|<=<|=|=|=|>|>=>|>>|>>=>|\\?|\\[|\\]|\\^|\\$|\\|||=|\\|||\\|~";this.↵
ASM={cN:"string",b:"'",e:'"',i:"\\n",c:["escape"],r:0};this.QSM={cN:"string",b:"'",e:'"',i:"\\n",c:["escape"],r:↵
:0};this.BE={cN:"escape",b:"\\\\",e:"^",nM:true,r:0};this.CLCM={cN:"comment",b:"/*",e:"$",r:0};this.CBLC↵
LM={cN:"comment",b:"/\\*",e:"\\/"};this.HCM={cN:"comment",b:"#",e:"$"};this.CNM={cN:"number",b:this.↵
CNR,e:"^",r:0}}()
— var initHighlightingOnLoad=hjs.initHighlightingOnLoad
— hjs XML_COMMENT={cN:"comment",b:"<!--",e:"-->"}
— hjs XML_ATTR={cN:"attribute",b:"\\s[A-Za-z0-9\\._-]+=",e:"^",c:["value"]}
— hjs XML_VALUE_QUOT={cN:"value",b:"'",e:'"' }
— hjs XML_VALUE_APOS={cN:"value",b:'"',e:"'"}
— hjs LANGUAGES xml={dM:{c:["pi","comment","cdata","tag"],cl:true,m:{cN:"pi",b:"<\\?",e:"\\?>",r:↵
:10},hjs.XML_COMMENT,{cN:"cdata",b:"<\\[CDATA\\[",e:"\\]\\>",{cN:"tag",b:"</?",e:">",c:["title","tag↵
_internal"],r:1.5},{cN:"title",b:"[A-Za-z0-9\\._-]+=",e:"^",r:0},{cN:"tag_internal",b:"^",eW:true,nM:true,c↵
:["attribute"],r:0,i:"[\\+\\.\\.]"},hjs.XML_ATTR,hjs.XML_VALUE_QUOT,hjs.XML_VALUE_APOS}}
— hjs HTML_TAGS={code:1,kbd:1,font:1,noscript:1,style:1,img:1,title:1,menu:1,tt:1,tr:1,param:1,li:1,tfoot↵
:1,th:1,input:1,td:1,dl:1,blockquote:1,fieldset:1,big:1,dd:1,abbr:1,optgroup:1,dt:1,button:1,isindex:1,p↵
:1,small:1,div:1,dir:1,em:1,frame:1,meta:1,sub:1,bdo:1,label:1,acronym:1,sup:1,body:1,xml:1,basefont↵
:1,base:1,br:1,address:1,strong:1,legend:1,ol:1,script:1,caption:1,s:1,col:1,h2:1,h3:1,h1:1,h6:1,h4:1,h5↵
:1,table:1,select:1,noframes:1,span:1,area:1,dfn:1,strike:1,cite:1,thead:1,head:1,option:1,form:1,hr:1,"var"↵
:1,link:1,b:1,colgroup:1,ul:1,applet:1,del:1,iframe:1,pre:1,frameset:1,ins:1,tbody:1,html:1,samp:1,map↵
:1,object:1,a:1,xmlns:1,center:1,textarea:1,i:1,q:1,u:1}
— hjs HTML_DOCTYPE={cN:"doctype",b:"<!DOCTYPE",e:">",r:10}
— hjs HTML_ATTR={cN:"attribute",b:"\\s[a-zA-Z\\_]+=",e:"^",c:["value"]}
— hjs HTML_SHORT_ATTR={cN:"attribute",b:"[a-zA-Z]+",e:"^"}
— hjs HTML_VALUE={cN:"value",b:"[a-zA-Z0-9]+",e:"^"}
— hjs LANGUAGES html={dM:{c:["tag","comment","doctype","vbscript"],cl:true,m:[hjs.XML_COMMENT,hjs.HTML_DOCTYPE ↵
N:"tag",l:[hjs.IR],k:hjs.HTML_TAGS,b:"<style",e:">",c:["attribute"],i:"[\\+\\.\\.]",starts:"css"},{cN:"tag",l↵
:[hjs.IR],k:hjs.HTML_TAGS,b:"<script",e:">",c:["attribute"],i:"[\\+\\.\\.]",starts:"javascript"},{cN:"tag",l:[hjs.↵
IR],k:hjs.HTML_TAGS,b:"<[A-Za-z]/",e:">",c:["attribute"],i:"[\\+\\.\\.]"},{cN:"css",e:"</style>",rE:true,sub↵
Language:"css"},{cN:"javascript",e:"</script>",rE:true,subLanguage:"javascript"},hjs.HTML_ATTR,hjs.HTML_SHORT_ATTR ↵
N:"vbscript",b:"<%",e:"%>",subLanguage:"vbscript"}}
— hjs LANGUAGES javascript={dM:{l:[hjs.UIR],c:["string","comment","number","regexp_container","function"],k↵
:{"keyword":{"in":1,"if":1,"for":1,"while":1,"finally":1,"var":1,"new":1,"function":1,"do":1,"return":1,"void"↵
:1,"else":1,"break":1,"catch":1,"instanceof":1,"with":1,"throw":1,"case":1,"default":1,"try":1,"this":1,"switch"↵
:1,"continue":1,"typeof":1,"delete":1},literal:{true:1,false:1,null:1}},m:[hjs.CLCM,hjs.CBLCML,hjs.C↵
NM,hjs.ASM,hjs.QSM,hjs.BE,{cN:"regexp_container",b:"("+hjs.RSR+"|case|return|throw)\\s*",e:"^",nM↵
:true,l:[hjs.IR],k:{"return":1,"throw":1,"case":1},c:["comment","regexp"],r:0},{cN:"regexp",b:"/.*?[^\\|\\|/][gim]*",e↵
:"^"},{cN:"function",b:"\\bfunction\\b",e:"{",l:[hjs.UIR],k:{"function":1},c:["title","params"]},{cN:"title",b:"[A-Za-↵
z$][0-9A-Za-z$]*",e:"^"},{cN:"params",b:"\\(",e:"\\)",c:["string","comment"]}}
— hjs LANGUAGES css={dM:{c:["at_rule","id","class","attr_selector","pseudo","rules","comment"],k:hjs.↵
HTML_TAGS,l:[hjs.IR],i:"=",cl:true,m:{cN:"at_rule",b:"@",e:"{",eE:true,l:[hjs.IR],k:{"import":1,page↵
:1,media:1,charset:1,"font-face":1},c:["function","string","number","pseudo"]},{cN:"id",b:"#[A-Za-z0-9_↵
-]+",e:"^"},{cN:"class",b:"\\.[A-Za-z0-9_]+",e:"^",r:0},{cN:"attr_selector",b:"\\[",e:"\\]",i:"$"},{cN:"pseudo",b:"↵

```

```
(:)?[a-zA-Z0-9\\_\\-\\+\\(\\)\\|\\\"\\'\\`]+",e:"^"},{cN:"rules",b:"{","e:"}"},c:["rule","comment"],i:["^\\s"]},{cN:"rule",b:"[A-Z\\_\\-\\+\\|\\\"\\'\\`]+\\s*:",e:";",eW:true,l:["[A-Za-z-]+"],k:{ "play-during":1,"counter-reset":1,"counter-increment":1,"min-height":1,"quotes":1,"border-top":1,"pitch":1,"font":1,"pause":1,"list-style-image":1,"border-width":1,"cue":1,"outline-width":1,"border-left":1,"elevation":1,"richness":1,"speech-rate":1,"border-bottom":1,"border-spacing":1,"background-color":1,"list-style-type":1,"text-align":1,"page-break-inside":1,"orphans":1,"page-break-before":1,"text-transform":1,"line-height":1,"padding-left":1,"font-size":1,"right":1,"word-spacing":1,"padding-top":1,"outline-style":1,"bottom":1,"content":1,"border-right-style":1,"padding-right":1,"border-left-style":1,"voice-family":1,"background-color":1,"border-bottom-color":1,"outline-color":1,"unicode-bidi":1,"max-width":1,"font-family":1,"caption-side":1,"border-right-width":1,"pause-before":1,"border-top-style":1,color:1,"border-collapse":1,"border-bottom-width":1,"float":1,height:1,"max-height":1,"margin-right":1,"border-top-width":1,speak:1,"speak-header":1,top:1,"cue-before":1,"min-width":1,width:1,"font-variant":1,"border-top-color":1,"background-position":1,"empty-cells":1,direction:1,"border-right":1,visibility:1,padding:1,"border-style":1,"background-attachment":1,overflow:1,"border-bottom-style":1,cursor:1,margin:1,display:1,"border-left-width":1,"letter-spacing":1,"vertical-align":1,clip:1,"border-color":1,"list-style":1,"padding-bottom":1,"pause-after":1,"speak-numeral":1,"margin-left":1,widows:1,border:1,"font-style":1,"border-left-color":1,"pitch-range":1,"background-repeat":1,"table-layout":1,"margin-bottom":1,"speak-punctuation":1,"font-weight":1,"border-right-color":1,"page-break-after":1,position:1,"white-space":1,"text-indent":1,"background-image":1,volume:1,stress:1,outline:1,clear:1,"z-index":1,"text-decoration":1,"margin-top":1,azimuth:1,"cue-after":1,left:1,"list-style-position":1},c:["value"]},hljs.CBLCML,{cN:"value",b:"^",eW:true,eE:true,c:["function","number","hexcolor","string","important","comment"],cN:"number",b:hljs.NR,e:"^"},{cN:"hexcolor",b:"\\#[0-9A-F]+",e:"^"},{cN:"function",b:hljs.IR+"\\(",e:"\\)"},c:["params"]},{cN:"params",b:"^",eW:true,eE:true,c:["number","string"]},{cN:"important",b:"!important",e:"^"},hljs.ASM,hljs.QSM}}
```

7.32.1 Documentation des variables

7.32.1.1 css

```
hljs LANGUAGES css ={dM:{c:["at_rule","id","class","attr_selector","pseudo","rules","comment"],k:hljs.HTML_TAGS,l:[hljs.IR],i:"="},cI:true,m:[{cN:"at_rule",b:"@",e:"[{}]",eE:true,l:[hljs.IR],k:{ "import":1,page:1,media:1,charset:1,"font-face":1},c:["function","string","number","pseudo"]},{cN:"id",b:"\\#[A-Za-z0-9_]+",e:"^"},{cN:"class",b:"\\. [A-Za-z0-9_]+",e:"^",r:0},{cN:"attr_selector",b:"\\[",e:"\\]",i:"$"},{cN:"pseudo",b:"(:)?[a-zA-Z0-9\\_\\-\\+\\(\\)\\|\\\"\\'\\`]+",e:"^"},{cN:"rules",b:"{","e:"}"},c:["rule","comment"],i:["^\\s"]},{cN:"rule",b:"[A-Z\\_\\-\\+\\|\\\"\\'\\`]+\\s*:",e:";",eW:true,l:["[A-Za-z-]+"],k:{ "play-during":1,"counter-reset":1,"counter-increment":1,"min-height":1,"quotes":1,"border-top":1,"pitch":1,"font":1,"pause":1,"list-style-image":1,"border-width":1,"cue":1,"outline-width":1,"border-left":1,"elevation":1,"richness":1,"speech-rate":1,"border-bottom":1,"border-spacing":1,"background-color":1,"list-style-type":1,"text-align":1,"page-break-inside":1,"orphans":1,"page-break-before":1,"text-transform":1,"line-height":1,"padding-left":1,"font-size":1,"right":1,"word-spacing":1,"padding-top":1,"outline-style":1,"bottom":1,"content":1,"border-right-style":1,"padding-right":1,"border-left-style":1,"voice-family":1,"background-color":1,"border-bottom-color":1,"outline-color":1,"unicode-bidi":1,"max-width":1,"font-family":1,"caption-side":1,"border-right-width":1,"pause-before":1,"border-top-style":1,color:1,"border-collapse":1,"border-bottom-width":1,"float":1,height:1,"max-height":1,"margin-right":1,"border-top-width":1,speak:1,"speak-header":1,top:1,"cue-before":1,"min-width":1,width:1,"font-variant":1,"border-top-color":1,"background-position":1,"empty-cells":1,direction:1,"border-right":1,visibility:1,padding:1,"border-style":1,"background-attachment":1,overflow:1,"border-bottom-style":1,cursor:1,margin:1,display:1,"border-left-width":1,"letter-spacing":1,"vertical-align":1,clip:1,"border-color":1,"list-style":1,"padding-bottom":1,"pause-after":1,"speak-numeral":1,"margin-left":1,widows:1,border:1,"font-style":1,"border-left-color":1,"pitch-range":1,"background-repeat":1,"table-layout":1,"margin-bottom":1,"speak-punctuation":1,"font-weight":1,"border-right-color":1,"page-break-after":1,position:1,"white-space":1,"text-indent":1,"background-image":1,volume:1,stress:1,outline:1,clear:1,"z-index":1,"text-decoration":1,"margin-top":1,azimuth:1,"cue-after":1,left:1,"list-style-position":1},c:["value"]},hljs.CBLCML,{cN:"value",b:"^",eW:true,eE:true,c:["function","number","hexcolor","string","important","comment"]},{cN:"number",b:hljs.NR,e:"^"},{cN:"hexcolor",b:"\\#[0-9A-F]+",e:"^"},{cN:"function",b:hljs.IR+"\\(",e:"\\)"},c:["params"]},{cN:"params",b:"^",eW:true,eE:true,c:["number","string"]},{cN:"important",b:"!important",e:"^"},hljs.ASM,hljs.QSM}}
```

```
N:"number",b:hljs.NR,e:"^"},{cN:"hexcolor",b:"\\#[0-9A-F]+",e:"^"},{cN:"function",b:hljs.I←
R+"\\(",e:"\\)",c:["params"]},{cN:"params",b:"^",eW:true,eE:true,c:["number","string"]},{cN←
:"important",b:"!important",e:"^"},hljs.ASM,hljs.QSM}]
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.2 hljs

```
var hljs = new function() { var q = {}; var a = {}; function o(c) { return c.replace(/&/gm, "&amp;").replace(/</gm, "&lt;";
l(t,s) { if (!t) { return false } for (var c=0; c<t.length; c++) { if (t[c]==s) { return true } } return false } function
e(t,s,c) { var u = "m" + (t.cI ? "i" : "") + (c ? "g" : ""); return new RegExp(s, u) } function k(s) { for (var c=0; c<s.←
childNodes.length; c++) { node = s.childNodes[c]; if (node.nodeName == "CODE") { return node } if (! (node.←
nodeType == 3 & node.nodeValue.match(/\\s+/))) { return null } } } function i(t) { var s = ""; for (var c=0; c<t.←
childNodes.length; c++) { if (t.childNodes[c].nodeType == 3) { s += t.childNodes[c].nodeValue } else { if (t.←
childNodes[c].nodeName == "BR") { s += "\\n" } else { s += i(t.childNodes[c]) } } } return s } function b(u) { var
s = u.className.split(/\\s+/); s = s.concat(u.parentNode.className.split(/\\s+/)); for (var c=0; c<s.←
length; c++) { var t = s[c].replace(/^language-/, ""); if (t == "no-highlight") { throw "No highlight" } if (q[t]) { return
t } } } function d(c) { var s = []; (function(u,v) { for (var t=0; t<u.childNodes.length; t++) { if (u.child←
Nodes[t].nodeType == 3) { v += u.childNodes[t].nodeValue.length } else { if (u.childNodes[t].nodeName == "BR") { v += 1 } else { s.←
push({ event: "start", offset: v, node: u.childNodes[t] }); v = arguments.callee(u.childNodes[t], v); s.←
push({ event: "stop", offset: v, node: u.childNodes[t] }) } } } return v) (c, 0); return s } function n(A,B,z) { var
t=0; var y=""; var v=[]; function w() { if (A.length & B.length) { if (A[0].offset != B[0].offset) { return (A[0].offset < B[0]
:B) } else { return (A[0].event == "start" & B[0].event == "stop") ? B:A } } else { return A.length ? A:B } } function
u(F) { var G = "<" + F.nodeName.toLowerCase(); for (var D=0; D<F.attributes.length; D++) { var E = F.←
attributes[D]; G += " " + E.nodeName.toLowerCase(); if (E.nodeValue != undefined) { G += "=" + o(E.node←
Value) + "'" } } return G + ">" } function C(D) { return "</" + D.nodeName.toLowerCase() + ">" } while (A.←
length || B.length) { var x = w().splice(0,1)[0]; y += o(z.substr(t,x.offset-t)); t = x.offset; if (x.←
event == "start") { y += u(x.node); v.push(x.node) } else { if (x.event == "stop") { var s = v.length; do { s--; var
c = v[s]; y += C(c) } while (c != x.node); v.splice(s,1); while (s < v.length) { y += u(v[s]); s++ } } } y += z.←
substr(t); return y } function h(K,E) { function L(Q,P) { Q.sm = []; for (var O=0; O<Q.c.length; O++) { for (var
N=0; N<P.m.length; N++) { if (P.m[N].cN == Q.c[O]) { Q.sm[Q.sm.length] = P.m[N] } } } } function A(N,P) { if (!P.←
c) { return null } if (!P.sm) { L(P,I) } for (var O=0; O<P.sm.length; O++) { if (P.sm[O].bR.test(N)) { return
P.sm[O] } } return null } function x(O,N) { if (D[O].e & D[O].eR.test(N)) { return 1 } if (D[O].eW) { var
P = x(O-1,N); return P ? P+1:0 } return 0 } function y(N,O) { return O.iR & O.iR.test(N) } function B(T,R) { var
P = []; function S(U) { if (!l(P,U)) { P[P.length] = U } if (T.c) { for (var O=0; O<R.m.length; O++) { if (l(T.←
c,R.m[O].cN)) { S(R.m[O].b) } } } var N = D.length-1; do { if (D[N].e) { S(D[N].e) } N-- } while (D[N+1].eW); if (T.←
i) { S(T.i) } var Q = "(" + P[0]; for (var O=0; O<P.length; O++) { Q += "|" + P[O] } Q += ")"; return e(R,Q) } function
t(P,O) { var Q = D[D.length-1]; if (!Q.t) { Q.t = B(Q,I) } P = P.substr(O); var N = Q.t.exec(P); if (!N) { return [P, "", true] } if (N.←
index == 0) { return ["", N[0], false] } else { return [P.substr(0, N.index), N[0], false] } } function s(R,N) { var
O = I.cI ? N[0].toLowerCase() : N[0]; for (var Q in R.keywordGroups) { if (!R.keywordGroups.hasOwnProperty(Q)) { continue } var P = R.keywordGroups[Q].hasOwnProperty(O); if (P) { return [Q, P] } } return
false } function G(Q,T) { if (!T.k || !T.l) { return o(Q) } if (!T.lR) { var S = "(" + T.l[0]; for (var P=1; P<T.←
l.length; P++) { S += "|" + T.l[P] } S += ")"; T.lR = e(I,S,true) } var R = ""; var U=0; T.lR.lastIndex=0; var O=T.←
lR.exec(Q); while (O) { R += o(Q.substr(U,O.index-U)); var N = s(T,O); if (N) { u += N[1]; R += "<span class=\"" + N[0] + "\">" + o(O[0]
lR.lastIndex; O = T.lR.exec(Q) } R += o(Q.substr(U,Q.length-U)); return R } function M(N,P) { if (P.sub←
Language & A[P.subLanguage]) { var O = h(P.subLanguage,N); u += O.keyword_count; C += O.r; return O.←
value } else { return G(N,P) } } function J(P,N) { var O = P.nM ? "" : "<span class=\"" + P.displayClassName + "\">"; if (P.←
rB) { c += O; P.buffer = "" } else { if (P.eB) { c += o(N) + O; P.buffer = "" } else { c += O; P.buffer = N } } D[D.length] = P } function
F(S,O,T) { var U = D[D.length-1]; if (T) { c += M(U.buffer+S,U); return false } var P = A(O,U); if (P) { c += M(U.←
buffer+S,U); J(P,O); C += P.r; return P.rB } var N = x(D.length-1,O); if (N) { var R = U.nM ? "" : "</span>"; if (U.←
rE) { c += M(U.buffer+S,U) + R } else { if (U.eE) { c += M(U.buffer+S,U) + R + o(O) } else { c += M(U.buffer+S+O,U) + R } } while (N > 1) { R = D[
length-2].nM ? "" : "</span>"; c += R; N--; D.length--; D[D.length-1].buffer = ""; if (U.starts) { for (var
Q=0; Q<I.m.length; Q++) { if (I.m[Q].cN == U.starts) { J(I.m[Q], ""); break } } } return U.rE } if (y(O,U)) { throw "Illegal" } var
I = q[K]; var D = [I.dM]; var C = 0; var u = 0; var c = ""; try { var w = 0; I.dM.buffer = ""; do { var z = t(E,w); var
```

```
v=F([z[0],z[1],z[2]]);w+=z[0].length;if(!v){w+=z[1].length;}while(!z[2]){throw"Illegal"}return{r:C,keyword_count:u,value:c}}catch(H){if(H=="Illegal"){return{r:0,keyword_count:0,value:o(E)}}}else{throw H}}}function j(){for(var s in q){if(!q.hasOwnProperty(s)){continue}var t=q[s];for(var c=0;c<t.m.length;c++){var u=t.m[c];if(u.b){u.bR=e(t,"^"+u.b)}if(u.e){u.eR=e(t,"^"+u.e)}if(u.i){u.iR=e(t,"^(?:"+u.i+")")}t.dM.iR=e(t,"^(?:"+t.dM.i+"")");if(u.r==undefined){u.r=1}if(!u.displayClassname){u.displayName=u.cN}}}function g(){function t(w){if(!w.keywordGroups){for(var v in w.k){if(!w.k.hasOwnProperty(v)){continue}if(w.k[v] instanceof Object){w.keywordGroups=w.k}else{w.keywordGroups={keyword:w.k}}break}}}for(var s in q){if(!q.hasOwnProperty(s)){continue}var u=q[s];t(u.dM);for(var c=0;c<u.m.length;c++){t(u.m[c])}}}function f(){if(f.called){return}f.called=true;j();g();a=qfunction r(y,C){f();try{var F=i(y);var v=b(y)}catch(z){if(z=="No highlight"){return}}B=h(v,F).valueelse{var D=0;for(var E in a){if(!a.hasOwnProperty(E)){continue}var t=h(E,F);var c=t.keyword_count+t.r;if(c>D){D=c;var B=t.value;v=E}}}if(B){var x=y.className;if(!x.match(v)){x+=" "+v}var s=d(y);if(s.length){var u=document.createElement("pre");u.innerHTML=B;B=n(s,d(u),F)}if(C){B=B.replace(/^((<[^>]+>|\t+)|gm,function(G,J,I,H){return J.replace(/\t/g,C)})}var A=document.createElement("div");A.innerHTML='<pre><code class="'+x+'">' + B + '</code></pre>';var w=y.parentNode.parentNode;w.replaceChild(A.firstChild,y.parentNode)}function m(){if(m.called){return}m.called=true;f();if(arguments.length){for(var c=0;c<arguments.length;c++){if(q[arguments[c]]){a[arguments[c]]=t=document.getElementsByTagName("pre");for(var c=0;c<t.length;c++){(var s=k(t[c]);if(s){r(s,hajs.tabReplace))}}}function p(){var c=arguments;var s=function() {m.apply(null,c)};if(window.addEventListener){window.addEventListener("DOMContentLoaded",s,false);window.addEventListener("load",s,false)}else attachEvent){window.attachEvent("onload",s)}else{window.onload=s}}this.LANGUAGES=q;this.initHighlightingOnLoadhighlightBlock=r;this.initHighlighting=m;this.IR="[a-zA-Z][a-zA-Z0-9_]*";this.UIR="[a-zA-Z_][a-zA-Z0-9_]*";this.NR="\b\d+(\.\d+)?";this.CNR="\b(0[xA-Za-z0-9]+\d+(\.\d+)?)";this.RSR=""!|!="||%=&|&&|&=\|*\|*=|\||\|=|,\|.|-|=|/|=|:<|<<|<=<|=|==|>=|>|>>|>=>|??|\[\]|\\\"ASM={cN:"string",b:'"',e:'',i:'\\n',c:["escape"],r:0};this.QSM={cN:"string",b:'"',e:'',i:'\\n',c:["escape"],r:0};this.BE={cN:"escape",b:'\\\\.',e:'^',nM:true,r:0};this.CLCM={cN:"comment",b:"//",e:"$",r:0};this.CBLCLM={cN:"comment",b:"\\/\\*",e:"\\*/"},this.HCM={cN:"comment",b:"# ",e:"$"};this.CNM={cN:"number",b:this.CNR,e:'^',r:0}}()
```

Définition à la ligne 1 du fichier `highlight.pack.js`.

7.32.1.3 html

```
hljs.LANGUAGES.html={dM:{c:["tag","comment","doctype","vbscript"],cI:true,m:[hljs.XML_COMMENT,hljs.HTML_DOCT
N:"tag",l:[hljs.IR],k:hljs.HTML_TAGS,b:"<style",e:">",c:["attribute"],i:"[\\+\\.]",starts←
:"css"},{cN:"tag",l:[hljs.IR],k:hljs.HTML_TAGS,b:"<script",e:">",c:["attribute"],i:"[\\+\\.]",starts←
:"javascript"},{cN:"tag",l:[hljs.IR],k:hljs.HTML_TAGS,b:"<[A-Za-z/]",e:">",c:["attribute"],i←
:"[\\+\\.]"},{cN:"css",e:"</style>",rE:true,subLanguage:"css"},{cN:"javascript",e:"</script>",r←
E:true,subLanguage:"javascript"},hljs.HTML_ATTR,hljs.HTML_SHORT_ATTR,hljs.XML_VALUE_QUOT,hljs.XML_VALUE_APOS,h
N:"vbscript",b:"<%",e:"%>",subLanguage:"vbscript"}]}
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.4 HTML_ATTR

```
hljs HTML_ATTR = {cN:"attribute",b:"\\s[a-zA-Z\\:\\_\\-]+=",e:"^",c:["value"]}
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.5 HTML_DOCTYPE

```
hljs HTML_DOCTYPE = {cN: "doctype", b: "<!DOCTYPE", e: ">", r: 10}
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.6 HTML_SHORT_ATTR

```
hljs HTML_SHORT_ATTR = {cN: "attribute", b: " [a-zA-Z]+", e: "^"}
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.7 HTML_TAGS

```
hljs HTML_TAGS = {code:1,kbd:1,font:1,noscript:1,style:1,img:1,title:1,menu:1,tt:1,tr:1,param↵
:1,li:1,tfoot:1,th:1,input:1,td:1,dl:1,blockquote:1,fieldset:1,big:1,dd:1,abbr:1,optgroup↵
:1,dt:1,button:1,isindex:1,p:1,small:1,div:1,dir:1,em:1,frame:1,meta:1,sub:1,bdo:1,label↵
:1,acronym:1,sup:1,body:1,xml:1,basefont:1,base:1,br:1,address:1,strong:1,legend:1,ol:1,script↵
:1,caption:1,s:1,col:1,h2:1,h3:1,h1:1,h6:1,h4:1,h5:1,table:1,select:1,noframes:1,span:1,area↵
:1,dfn:1,strike:1,cite:1,thead:1,head:1,option:1,form:1,hr:1,"var":1,link:1,b:1,colgroup↵
:1,ul:1,applet:1,del:1,iframe:1,pre:1,frameset:1,ins:1,tbody:1,html:1,samp:1,map:1,object↵
:1,a:1,xmlns:1,center:1,textarea:1,i:1,q:1,u:1}
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.8 HTML_VALUE

```
hljs HTML_VALUE = {cN: "value", b: " [a-zA-Z0-9]+", e: "^"}
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.9 initHighlightingOnLoad

```
var initHighlightingOnLoad = hljs.initHighlightingOnLoad
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.10 javascript

```
hljs LANGUAGES javascript =({dm:{l:[hljs.UIR],c:["string","comment","number","regex_container","function"],k:
:{"keyword":{"in":1,"if":1,"for":1,"while":1,"finally":1,"var":1,"new":1,"function":1,"do":1,
:1,"return":1,"void":1,"else":1,"break":1,"catch":1,"instanceof":1,"with":1,"throw":1,"case":1,
:1,"default":1,"try":1,"this":1,"switch":1,"continue":1,"typeof":1,"delete":1},literal:{true:1,
:1,"false":1,"null":1}}},m:[hljs.CLCM,hljs.CBLCM,hljs.CNM,hljs.ASM,hljs.QSM,hljs.BE,{c:
N:"regex_container",b:"("+hljs.RSR+"|case|return|throw)\\s*",e:"^",nM:true,l:[hljs.IR],k:
:{"return":1,"throw":1,"case":1},c:["comment","regex"],r:0},{cN:"regex",b:"/.+?[^\n\r]/[gim]*",e:
:"^"},{cN:"function",b:"\\bfunction\\b",e:"{",l:[hljs.UIR],k:{function:1},c:["title","params"]},{c:
N:"title",b:"[A-Za-z$_][0-9A-Za-z$_]*",e:"^"},{cN:"params",b:"\\(",e:"\\)",c:["string","comment"]}}})
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.11 xml

```
hljs LANGUAGES xml =({dm:{c:["pi","comment","cdata","tag"],cI:true,m:[{cN:"pi",b:"<\\?",e:
:"\\?>",r:10},hljs.XML_COMMENT,{cN:"cdata",b:"<\\!\\[CDATA\\[",e:"\\]\\]>"},{cN:"tag",b:
:"</?",e:">",c:["title","tag_internal"],r:1.5},{cN:"title",b:"[A-Za-z0-9\\.\\_\\-\\+]",e:"^",r:
:0},{cN:"tag_internal",b:"^",eW:true,nM:true,c:["attribute"],r:0,i:"[\\+\\.\\.]"},hljs.XML_ATTR,hljs.XML_VALUE_Q
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.12 XML_ATTR

```
hljs XML_ATTR =({cN:"attribute",b:"\\s[A-Za-z0-9\\.\\_\\-\\+]=",e:"^",c:["value"]})
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.13 XML_COMMENT

```
hljs XML_COMMENT =({cN:"comment",b:"<!--",e:"-->"})
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.14 XML_VALUE_APOS

```
hljs XML_VALUE_APOS =({cN:"value",b:"'",e:"'"})
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.32.1.15 XML_VALUE_QUOT

```
hljs XML_VALUE_QUOT = {cN:"value",b:'"',e:'"'}
```

Définition à la ligne 1 du fichier highlight.pack.js.

7.33 Référence du fichier JS/jquery-Rotate-Plugin.js

Variables

— fn [rotate](#)

7.33.1 Documentation des variables

7.33.1.1 rotate

```
fn rotate
```

Définition à la ligne 1 du fichier jQuery-Rotate-Plugin.js.

7.34 Référence du fichier JS/jquery.easing.1.3.js

Fonctions

— jQuery [extend](#) (jQuery.easing, { def:'easeOutQuad', swing:function(x, t, b, c, d) { return jQuery.easing[jQuery.easing.def](x, t, b, c, d);}, easeInQuad:function(x, t, b, c, d) { return c * (t/=d) * t+b;}, easeOutQuad:function(x, t, b, c, d) { return -c * (t/=d) * (t-2)+b;}, easeInOutQuad:function(x, t, b, c, d) { if((t/=d/2)< 1) return c/2 * t * t+b;return -c/2 * ((-t) * (t-2) - 1)+b;}, easeInCubic:function(x, t, b, c, d) { return c * (t/=d) * t * t+b;}, easeOutCubic:function(x, t, b, c, d) { return c * ((t/=d-1) * t * t+1)+b;}, easeInOutCubic:function(x, t, b, c, d) { if((t/=d/2)< 1) return c/2 * t * t * t+b;return c/2 * ((t=2) * t * t+2)+b;}, easeInQuart:function(x, t, b, c, d) { return c * (t/=d) * t * t * t+b;}, easeOutQuart:function(x, t, b, c, d) { return -c * ((t/=d-1) * t * t * t - 1)+b;}, easeInOutQuart:function(x, t, b, c, d) { if((t/=d/2)< 1) return c/2 * t * t * t * t+b;return -c/2 * ((t=2) * t * t * t - 2)+b;}, easeInQuint:function(x, t, b, c, d) { return c * (t/=d) * t * t * t * t+b;}, easeOutQuint:function(x, t, b, c, d) { return c * ((t/=d-1) * t * t * t * t+1)+b;}, easeInOutQuint:function(x, t, b, c, d) { if((t/=d/2)< 1) return c/2 * t * t * t * t * t+b;return c/2 * ((t=2) * t * t * t * t+2)+b;}, easeInSine:function(x, t, b, c, d) { return -c * Math.cos(t/d * (Math.PI/2))+c+b;}, easeOutSine:function(x, t, b, c, d) { return c * Math.sin(t/d * (Math.PI/2))+b;}, easeInOutSine:function(x, t, b, c, d) { return -c/2 * (Math.cos(Math.PI * t/d) - 1)+b;}, easeInExpo:function(x, t, b, c, d) { return (t==0) ? b : c * Math.pow(2, 10 * (t/d - 1))+b;}, easeOutExpo:function(x, t, b, c, d) { return (t==d) ? b+c : c * (-Math.pow(2, -10 * (t/d)+1))+b;}, easeInOutExpo:function(x, t, b, c, d) { if(t==0) return b;if(t==d) return b+c;if((t/=d/2)< 1) return c/2 * Math.pow(2, 10 * (t - 1))+b;return c/2 * (-Math.pow(2, -10 * --t)+2)+b;}, easeInCirc:function(x, t, b, c, d) { return -c * (Math.sqrt(1 - (t/=d) * t) - 1)+b;}, easeOutCirc:function(x, t, b, c, d) { return c * Math.sqrt(1 - (t/=d-1) * t)+b;}, easeInOutCirc:function(x, t, b, c, d) { if((t/=d/2)< 1) return -c/2 * (Math.sqrt(1 - t * t) - 1)+b;return c/2 * (Math.sqrt(1 - (t=2) * t)+1)+b;}, easeInElastic:function(x, t, b, c, d) { var s=1.70158;var p=0;var a=c;if(t==0) return b;if((t/=d)==1) return b+c;if(!p) p=d *.3;if(a< Math.abs(c)) { a=c;var s=p/4;} [else](#) var s=p/(2 * Math.PI) * Math.asin(c/a);return -(a * Math.pow(2, 10 * (t=1)) * Math.sin((t * d-s) * (2 * Math.PI)/p))+b;}, easeOutElastic:function(x, t, b, c, d) { var s=1.70158;var p=0;var a=c;if(t==0) return b;if((t/=d)==1) return b+c;if(!p) p=d *.3;if(a< Math.abs(c)) { a=c;var s=p/4;} [else](#) var s=p/(2 * Math.PI) * Math.asin(c/a);return

```

a*Math.pow(2,-10*t)*Math.sin((t*d-s)*(2*Math.PI)/p)+c+b;}, easeInOutElastic:function(x, t, b, c, d) { var
s=1.70158;var p=0;var a=c;if(t==0) return b;if((t/d/2)==2) return b+c;if(!p) p=d*(.3*1.5);if(a< Math.abs(c))
{ a=c;var s=p/4;} else var s=p/(2*Math.PI)*Math.asin(c/a);if(t< 1) return -.5*(a*Math.pow(2, 10*(t=1))
*Math.sin((t*d-s)*(2*Math.PI)/p))+b;return a*Math.pow(2,-10*(t=1))*Math.sin((t*d-s)*(2*Math.PI)/p)
*.5+c+b;}, easeInBack:function(x, t, b, c, d, s) { if(s==undefined) s=1.70158;return c*(t/d)*t*((s+1)*t
-s)+b;}, easeOutBack:function(x, t, b, c, d, s) { if(s==undefined) s=1.70158;return c*((t/d-1)*t*((s+1)
*t+s)+1)+b;}, easeInOutBack:function(x, t, b, c, d, s) { if(s==undefined) s=1.70158;if((t/d/2)< 1) return c/2
*(t*t*((s*(1.525))+1)*t-s))+b;return c/2*((t=2)*t*((s*(1.525))+1)*t+s)+2)+b;}, easeInBounce:
function(x, t, b, c, d) { return c - jQuery.easing.easeOutBounce(x, d-t, 0, c, d)+b;}, easeOutBounce:function(x,
t, b, c, d) { if((t/d)<(1/2.75)) { return c*(7.5625*t*t)+b;} else if(t<(2/2.75)) { return c*(7.5625*(t=(1.
5/2.75))*t+.75)+b;} else if(t<(2.5/2.75)) { return c*(7.5625*(t=(2.25/2.75))*t+.9375)+b;} else { return c
*(7.5625*(t=(2.625/2.75))*t+.984375)+b;} }, easeInOutBounce:function(x, t, b, c, d) { if(t< d/2) return j
Query.easing.easeInBounce(x, t*2, 0, c, d)*.5+b;return jQuery.easing.easeOutBounce(x, t*2-d, 0, c, d)
*.5+c*.5+b;}})

```

Variables

— `jQuery.easing` [`'swing'`] = `jQuery.easing['swing']`

7.34.1 Documentation des fonctions

7.34.1.1 extend()

```

jQuery.extend (
    jQuery. easing,
    { def:'easeOutQuad', swing:function(x, t, b, c, d) { return jQuery.easing[jQuery.easing.def](x, t, b, c, d);}, easeInQuad:function(x, t, b, c, d) { return c*(t/=d)*t+b;}, easeOutQuad:function(x, t, b, c, d) { return -c*(t/=d)*(t-2)+b;}, easeInOutQuad:
function(x, t, b, c, d) { if((t/=d/2)< 1) return c/2*t*t+b;return -c/2*((t--)*t*(t-2) - 1)+b;}, easeInCubic:function(x, t, b, c, d) { return c*(t/=d)*t*t+b;}, easeOutCubic:function(x,
t, b, c, d) { return c*((t=t/d-1)*t*t+1)+b;}, easeInOutCubic:function(x, t, b, c, d) { if((t/=d/2)< 1) return c/2*t*t*t+b;return c/2*((t--2)*t*t*(t+2)+b;}, easeInQuart:function(x, t, b, c, d)
{ return c*(t/=d)*t*t*t+b;}, easeOutQuart:function(x, t, b, c, d) { return -c*((t=t/d-1)*t*t*t - 1)+b;}, easeInOutQuart:function(x, t, b, c, d) { if((t/=d/2)< 1) return c/2*t*t*t*t+b;return -c/2*((t--2)*t*t*t*t - 2)+b;}, easeInQuint:function(x, t, b, c, d) { return
c*(t/=d)*t*t*t*t+b;}, easeOutQuint:function(x, t, b, c, d) { return c*((t=t/d-1)*t*t*t*t+1)+b;}, easeInOutQuint:function(x, t, b, c, d) { if((t/=d/2)< 1) return c/2*t*t*t*t*t+b;return c/2*((t--2)*t*t*t*t*(t+2)+b;}, easeInSine:function(x, t, b, c, d) { return -c
*Math.cos(t/d*(Math.PI/2))+c+b;}, easeOutSine:function(x, t, b, c, d) { return c*Math.sin(t/d*(Math.PI/2))+b;}, easeInOutSine:function(x, t, b, c, d) { return -c/2*(Math.cos(Math.PI
*t/d) - 1)+b;}, easeInExpo:function(x, t, b, c, d) { return(t==0)? b:c*Math.pow(2, 10*(t/d - 1))+b;}, easeOutExpo:function(x, t, b, c, d) { return(t==d)? b+c:c*(-Math.pow(2, -10
*t/d)+1)+b;}, easeInOutExpo:function(x, t, b, c, d) { if(t==0) return b;if(t==d) return b+c;if((t/=d/2)< 1) return c/2*Math.pow(2, 10*(t - 1))+b;return c/2*(-Math.pow(2, -10*--t)+2)+b;}, easeInCirc:
function(x, t, b, c, d) { return -c*(Math.sqrt(1-(t/=d)*t) - 1)+b;}, easeOutCirc:
function(x, t, b, c, d) { return c*Math.sqrt(1-(t=t/d-1)*t)+b;}, easeInOutCirc:function(x,
t, b, c, d) { if((t/=d/2)< 1) return -c/2*(Math.sqrt(1-t*t) - 1)+b;return c/2*(Math.sqrt(1-(t--2)*t)+1)+b;}, easeInElastic:function(x, t, b, c, d) { var s=1.70158;var p=0;var
a=c;if(t==0) return b;if((t/=d)==1) return b+c;if(!p) p=d*.3;if(a< Math.abs(c)) { a=c;var
s=p/4;} else var s=p/(2*Math.PI)*Math.asin(c/a);return -(a*Math.pow(2, 10*(t--1))*Math.sin((t*d-s)*(2*Math.PI)/p))+b;}, easeOutElastic:function(x, t, b, c, d) { var s=1.70158;var

```

```

p=0;var a=c;if(t==0) return b;if((t/=d)==1) return b+c;if(!p) p=d*.3;if(a< Math.abs(c)) {
a=c;var s=p/4;} else var s=p/(2*Math.PI)*Math.asin(c/a);return a*Math.pow(2,-10*t)*Math.↵
sin((t*d-s)*(2*Math.PI)/p)+c+b;}, easeInOutElastic:function(x, t, b, c, d) { var s=1.↵
70158;var p=0;var a=c;if(t==0) return b;if((t/=d/2)==2) return b+c;if(!p) p=d*(.3*1.5);if(a<
Math.abs(c)) { a=c;var s=p/4;} else var s=p/(2*Math.PI)*Math.asin(c/a);if(t< 1) return -.↵
5*(a*Math.pow(2, 10*(t-=1))*Math.sin((t*d-s)*(2*Math.PI)/p))+b;return a*Math.pow(2,-10
*(t-=1))*Math.sin((t*d-s)*(2*Math.PI)/p)*.5+c+b;}, easeInBack:function(x, t, b, c, d, s) {
if(s==undefined) s=1.70158;return c*(t/=d)*t*((s+1)*t - s)+b;}, easeOutBack:function(x, t,
b, c, d, s) { if(s==undefined) s=1.70158;return c*((t=t/d-1)*t*((s+1)*t+s)+1)+b;}, ease↵
InOutBack:function(x, t, b, c, d, s) { if(s==undefined) s=1.70158;if((t/=d/2)< 1) return c/2
*(t*t*((s*=(1.525))+1)*t - s))+b;return c/2*((t-=2)*t*((s*=(1.525))+1)*t+s)+2)+b;},
easeInBounce:function(x, t, b, c, d) { return c - jQuery.easing.easeOutBounce(x, d-t, 0, c,
d)+b;}, easeOutBounce:function(x, t, b, c, d) { if((t/=d)<(1/2.75)) { return c*(7.5625*t
*t)+b;} else if(t<(2/2.75)) { return c*(7.5625*(t-=(1.5/2.75))*t+.75)+b;} else if(t<(2.↵
5/2.75)) { return c*(7.5625*(t-=(2.25/2.75))*t+.9375)+b;} else { return c*(7.5625*(t-=(2.↵
625/2.75))*t+.984375)+b;} }, easeInOutBounce:function(x, t, b, c, d) { if(t< d/2) return j↵
Query.easing.easeInBounce(x, t*2, 0, c, d)*.5+b;return jQuery.easing.easeOutBounce(x, t*2-d,
0, c, d)*.5+c*.5+b;}} )

```

7.34.2 Documentation des variables

7.34.2.1 easing

```
jQuery.easing['jswing'] = jQuery.easing['swing']
```

Définition à la ligne 39 du fichier jquery.easing.1.3.js.

7.35 Référence du fichier JS/jquery.modal.js

Fonctions

- [function](#) (factory)
- [function](#) (\$, window, document, undefined)

7.35.1 Documentation des fonctions

7.35.1.1 function() [1/2]

```
function (
    factory
)
```

Définition à la ligne 6 du fichier jquery.modal.js.

7.35.1.2 function() [2/2]

```
function (
    $,
    window ,
    document ,
    undefined )
```

Définition à la ligne 15 du fichier jquery.modal.js.

7.36 Référence du fichier JS/sup_perso.js

Ajax pour la suppression de personnage.

Fonctions

- `function auto_load ()`
- `sup_perso submit (function() { s=$(this).serialize();$.ajax({ type:"POST", data:s, url:'ajax/sup_perso.php', success:function(retour){ auto_load();} });return false;})`

7.36.1 Description détaillée

Ajax pour la suppression de personnage.

Auteur

Guillaume Nagiel

Version

1.0

Date

5 Mars 2018

Ce fichier js sert à supprimer un personnage du deck via une requete Ajax.

7.36.2 Documentation des fonctions

7.36.2.1 auto_load()

```
function auto_load ( )
```

Définition à la ligne 15 du fichier sup_perso.js.

7.36.2.2 submit()

```
sup_perso submit (
    function() { s=$(this).serialize();$.ajax({ type:"POST", data:s, url:'ajax/sup_↵
perso.php', success:function(retour){ auto_load();} });return false;} )
```

7.37 Référence du fichier JS/team.js

Ajax pour gestion de l'équipe.

Fonctions

- `function auto_load ()`
- `add_team submit (function() { s=$(this).serialize();$.ajax({ type:"POST", data:s, url:'ajax/team_ajax.php', success:function(retour){ auto_load();} });return false;})`

7.37.1 Description détaillée

Ajax pour gestion de l'équipe.

Auteur

Guillaume Nagiel

Version

1.0

Date

5 Mars 2018

Ce fichier js sert à modifier l'état de l'équipe via une requete Ajax.

7.37.2 Documentation des fonctions

7.37.2.1 auto_load()

```
function auto_load ( )
```

Définition à la ligne 15 du fichier team.js.

7.37.2.2 submit()

```
add_team submit (
    function() { s=$(this).serialize();$.ajax({ type:"POST", data:s, url:'ajax/team_↵
ajax.php', success:function(retour){ auto_load();} });return false;} )
```

Index

\$Xp_Max
 User, [91](#)

\$argent
 User, [90](#)

\$arm
 Personnage, [52](#)

\$atout
 Personnage, [52](#)

\$atr_1
 Stuff, [73](#)

\$atr_2
 Stuff, [73](#)

\$att
 Base_personnage, [24](#)

\$cout
 Stuff, [73](#)

\$crit
 Personnage, [52](#)

\$def
 Personnage, [52](#)

\$descr
 Stuff, [73](#)

\$email
 User, [90](#)

\$emp_team
 Base_personnage, [24](#)

\$emp_team_max
 User, [90](#)

\$endu
 User, [90](#)

\$endu_Max
 User, [90](#)

\$entry
 gestion_up_perso.php, [99](#)

\$esc
 Personnage, [53](#)

\$etat
 Personnage, [53](#)

\$etoile
 Base_personnage, [24](#)
 Personnage, [53](#)
 Stuff, [74](#)

\$id_base
 Stuff, [74](#)

\$id_perso
 Personnage, [53](#)

\$id_persos
 Base_personnage, [24](#)

\$id_stuff
 Stuff, [74](#)

\$id_user
 Personnage, [53](#)
 User, [90](#)

\$json
 gestion_up_perso.php, [100](#)

\$manager
 add_perso.php, [95](#)
 gestion_stuff.php, [97](#)
 gestion_up_perso.php, [100](#)
 sup_perso.php, [101](#)
 team_ajax.php, [102](#)

\$managerStuff
 add_perso.php, [96](#)
 gestion_stuff.php, [98](#)

\$managerUser
 gestion_stuff.php, [98](#)

\$mdp
 User, [91](#)

\$niveau
 Personnage, [53](#)
 Stuff, [74](#)
 User, [91](#)

\$nom
 Base_personnage, [24](#)
 Stuff, [74](#)

\$or_
 User, [91](#)

\$pen
 Personnage, [54](#)

\$perso
 add_perso.php, [96](#)
 gestion_stuff.php, [98](#)
 gestion_up_perso.php, [100](#)
 sup_perso.php, [101](#)

\$pre
 Personnage, [54](#)

\$pseudo
 User, [91](#)

\$pv
 Base_personnage, [24](#)

\$qualite
 Base_personnage, [25](#)
 Personnage, [54](#)
 Stuff, [74](#)

\$soi
 Personnage, [54](#)

\$stuff
 gestion_stuff.php, [98](#)

- \$stuff_1
 - Base_personnage, 25
- \$stuff_2
 - Base_personnage, 25
- \$stuff_3
 - Base_personnage, 25
- \$stuff_4
 - Base_personnage, 25
- \$tab
 - gestion_stuff.php, 98
- \$team
 - Personnage, 54
- \$ten
 - Personnage, 54
- \$timeEndormi
 - Personnage, 55
- \$types
 - Base_personnage, 26
 - Stuff, 75
- \$user
 - add_perso.php, 96
- \$usermanager
 - add_perso.php, 96
- \$vit
 - Personnage, 55
- \$xp
 - Personnage, 55
 - User, 91
- \$xp_max
 - Personnage, 55
- __construct
 - Base_personnage, 15
 - Personnage, 29
 - PersonnagesManager, 57
 - Stuff, 63
 - StuffManager, 75
 - User, 79
 - UserManager, 92
- add
 - PersonnagesManager, 58
 - StuffManager, 76
 - UserManager, 93
- add_perso.js
 - auto_load, 112
 - submit, 112
- add_perso.php
 - \$manager, 95
 - \$managerStuff, 96
 - \$perso, 96
 - \$user, 96
 - \$usermanager, 96
 - chargerClasse, 95
- affichePerso
 - Function_Jeu.php, 107
- ajax/add_perso.php, 95
- ajax/gestion_perso.php, 96
- ajax/gestion_stuff.php, 97
- ajax/gestion_sup_tab.php, 99
- ajax/gestion_up_perso.php, 99
- ajax/sup_perso.php, 100
- ajax/team_ajax.php, 101
- argent
 - User, 80
- arm
 - Personnage, 29
- atout
 - Personnage, 30
- atr_1
 - Stuff, 63
- atr_2
 - Stuff, 64
- att
 - Base_personnage, 16
- Att_soi, 12
 - lancerUnSoin, 13
- auto_load
 - add_perso.js, 112
 - sup_perso.js, 123
 - team.js, 124
- Base_personnage, 13
 - \$att, 24
 - \$emp_team, 24
 - \$etoile, 24
 - \$id_persos, 24
 - \$nom, 24
 - \$pv, 24
 - \$qualite, 25
 - \$stuff_1, 25
 - \$stuff_2, 25
 - \$stuff_3, 25
 - \$stuff_4, 25
 - \$types, 26
 - __construct, 15
 - att, 16
 - etoile, 16
 - hydrate, 16
 - id_persos, 16
 - nom, 17
 - pv, 17
 - qualite, 17
 - setAtt, 18
 - setEmp_team, 18
 - setId_persos, 18
 - setNom, 19
 - setPv, 19
 - setQualite, 20
 - setStuff_1, 20
 - setStuff_2, 20
 - setStuff_3, 21
 - setStuff_4, 21
 - setTypes, 21
 - stuff_1, 22
 - stuff_2, 22
 - stuff_3, 22
 - stuff_4, 23
 - types, 23

- CEST_MOI
 - Personnage, 55
- cadre_team
 - team.php, 109
- chargerClass
 - index.php, 111
- chargerClasse
 - add_perso.php, 95
 - gestion_stuff.php, 97
 - gestion_up_perso.php, 99
 - sup_perso.php, 101
 - team_ajax.php, 102
- class/Att_soi.php, 102
- class/Base_personnage.php, 102
- class/Personnage.php, 103
- class/PersonnagesManager.php, 103
- class/Soigneur.php, 103
- class/Stuff.php, 103
- class/StuffManager.php, 104
- class/User.php, 104
- class/UserManager.php, 104
- count
 - PersonnagesManager, 58
 - StuffManager, 76
 - UserManager, 93
- cout
 - Stuff, 64
- crit
 - Personnage, 30
- css
 - highlight.pack.js, 115
- debut_combat
 - Personnage, 31
- def
 - Personnage, 31
- delete
 - PersonnagesManager, 58
 - StuffManager, 76
 - UserManager, 93
- descr
 - Stuff, 65
- easing
 - jquery.easing.1.3.js, 122
- else
 - gestion_perso.php, 97
 - gestion_stuff.php, 98
- email
 - User, 80
- emp_team_max
 - User, 81
- endu
 - User, 81
- endu_Max
 - User, 81
- esc
 - Personnage, 32
- estEndormi
 - Personnage, 32
- etat
 - Personnage, 33
- etoile
 - Base_personnage, 16
 - Personnage, 33
 - Stuff, 65
- exists
 - PersonnagesManager, 58
 - StuffManager, 76
 - UserManager, 93
- extend
 - jquery.easing.1.3.js, 121
- fonctionComparaison
 - Function_Jeu.php, 107
- frapper
 - Personnage, 33
- function
 - jquery.modal.js, 122
- Function_Jeu.php
 - affichePerso, 107
 - fonctionComparaison, 107
- gestion_perso.php
 - else, 97
- gestion_stuff.php
 - \$manager, 97
 - \$managerStuff, 98
 - \$managerUser, 98
 - \$perso, 98
 - \$stuff, 98
 - \$tab, 98
 - chargerClasse, 97
 - else, 98
- gestion_up_perso.php
 - \$entry, 99
 - \$json, 100
 - \$manager, 100
 - \$perso, 100
 - chargerClasse, 99
- get
 - PersonnagesManager, 58
 - StuffManager, 77
 - UserManager, 93
- get_base_perso
 - PersonnagesManager, 58
 - StuffManager, 77
- get_id_team
 - PersonnagesManager, 59
- get_niveau_sup
 - StuffManager, 77
- getList
 - PersonnagesManager, 59
 - StuffManager, 77
 - UserManager, 93
- getList_up
 - PersonnagesManager, 59
- getListCheat

- PersonnagesManager, 59
- StuffManager, 77
- HTML_ATTR
 - highlight.pack.js, 117
- HTML_DOCTYPE
 - highlight.pack.js, 117
- HTML_SHORT_ATTR
 - highlight.pack.js, 118
- HTML_TAGS
 - highlight.pack.js, 118
- HTML_VALUE
 - highlight.pack.js, 118
- highlight.pack.js
 - css, 115
 - HTML_ATTR, 117
 - HTML_DOCTYPE, 117
 - HTML_SHORT_ATTR, 118
 - HTML_TAGS, 118
 - HTML_VALUE, 118
 - hljs, 116
 - html, 117
 - initHighlightingOnLoad, 118
 - javascript, 118
 - XML_ATTR, 119
 - XML_COMMENT, 119
 - XML_VALUE_APOS, 119
 - XML_VALUE_QUOT, 119
 - xml, 119
- hljs
 - highlight.pack.js, 116
- html
 - highlight.pack.js, 117
- hydrate
 - Base_personnage, 16
 - Personnage, 34
 - Stuff, 65
 - User, 82
- id_base
 - Stuff, 66
- id_perso
 - Personnage, 35
- id_persos
 - Base_personnage, 16
- id_stuff
 - Stuff, 66
- id_user
 - Personnage, 35
 - User, 82
- includes/Function_Attaque.php, 106
- includes/Function_Jeu.php, 106
- includes/cheat.php, 104
- includes/cheat/modif_stuff.php, 105
- includes/cheat/modif_team.php, 105
- includes/eviterMessageAvertissement.php, 106
- includes/identifiant.php, 108
- includes/stuff.php, 104
- includes/stuff_modal.php, 108
- includes/team - save.php, 109
- includes/team.php, 109
- includes/up_perso.php, 110
- includes/up_perso_modal.php, 110
- incrementAtout
 - Personnage, 35
- index.php, 111
 - chargerClass, 111
- initHighlightingOnLoad
 - highlight.pack.js, 118
- jQuery-Rotate-Plugin.js
 - rotate, 120
- JS/add_perso.js, 112
- JS/highlight/highlight.pack.js, 113
- JS/jQuery-Rotate-Plugin.js, 120
- JS/jquery.easing.1.3.js, 120
- JS/jquery.modal.js, 122
- JS/sup_perso.js, 123
- JS/team.js, 124
- javascript
 - highlight.pack.js, 118
- jquery.easing.1.3.js
 - easing, 122
 - extend, 121
- jquery.modal.js
 - function, 122
- lancerUnSoin
 - Att_soi, 13
 - Soigneur, 62
- lvlUp
 - Stuff, 66
- mdp
 - User, 82
- niveau
 - Personnage, 36
 - Stuff, 67
 - User, 83
- nom
 - Base_personnage, 17
 - Stuff, 67
- nomValide
 - User, 83
- or_
 - User, 83
- PAS_DE_MAGIE
 - Personnage, 55
- PERSO_ENDORMI
 - Personnage, 56
- PERSO_FULL_LIFE
 - Personnage, 56
- PERSONNAGE_ENSORCELE
 - Personnage, 56
- PERSONNAGE_FRAPPE
 - Personnage, 56

PERSONNAGE_SOIGNE
 Personnage, 56
PERSONNAGE_TUE
 Personnage, 56
payer_argent
 User, 84
pen
 Personnage, 36
Personnage, 26
 \$arm, 52
 \$atout, 52
 \$crit, 52
 \$def, 52
 \$esc, 53
 \$etat, 53
 \$etoile, 53
 \$id_perso, 53
 \$id_user, 53
 \$niveau, 53
 \$pen, 54
 \$pre, 54
 \$qualite, 54
 \$soi, 54
 \$team, 54
 \$ten, 54
 \$timeEndormi, 55
 \$vit, 55
 \$xpr, 55
 \$xpr_max, 55
 __construct, 29
 arm, 29
 atout, 30
 CEST_MOI, 55
 crit, 30
 debut_combat, 31
 def, 31
 esc, 32
 estEndormi, 32
 etat, 33
 etoile, 33
 frapper, 33
 hydrate, 34
 id_perso, 35
 id_user, 35
 incrementAtout, 35
 niveau, 36
 PAS_DE_MAGIE, 55
 PERSO_ENDORMI, 56
 PERSO_FULL_LIFE, 56
 PERSONNAGE_ENSORCELE, 56
 PERSONNAGE_FRAPPE, 56
 PERSONNAGE_SOIGNE, 56
 PERSONNAGE_TUE, 56
 pen, 36
 pre, 37
 qualite, 37
 recevoirDegats, 37
 recevoirSoins, 38
 recevoirXP, 38
 reveil, 39
 setArm, 39
 setAtout, 40
 setCrit, 40
 setDef, 41
 setEsc, 41
 setEtat, 41
 setEtoile, 42
 setId_perso, 42
 setId_user, 43
 setNiveau, 43
 setPen, 43
 setPre, 44
 setQualite, 44
 setSoi, 45
 setTeam, 45
 setTen, 45
 setTimeEndormi, 46
 setTypes, 46
 setVit, 47
 setXp, 47
 setXp_max, 47
 soi, 48
 soigner, 48
 team, 49
 ten, 49
 timeEndormi, 50
 types, 50
 vit, 51
 xp, 51
 xp_max, 51
PersonnagesManager, 57
 __construct, 57
 add, 58
 count, 58
 delete, 58
 exists, 58
 get, 58
 get_base_perso, 58
 get_id_team, 59
 getList, 59
 getList_up, 59
 getListCheat, 59
 setDb, 59
 sup_team, 60
 update, 60
 update_team, 60
pre
 Personnage, 37
pseudo
 User, 84
pv
 Base_personnage, 17
qualite
 Base_personnage, 17
 Personnage, 37
 Stuff, 68

recevoirDegats
 Personnage, 37

recevoirSoins
 Personnage, 38

recevoirXP
 Personnage, 38

reveil
 Personnage, 39

rotate
 jQuery-Rotate-Plugin.js, 120

setArgent
 User, 84

setArm
 Personnage, 39

setAtout
 Personnage, 40

setAtr_1
 Stuff, 68

setAtr_2
 Stuff, 68

setAtt
 Base_personnage, 18

setCout
 Stuff, 69

setCrit
 Personnage, 40

setDb
 PersonnagesManager, 59
 StuffManager, 77
 UserManager, 94

setDef
 Personnage, 41

setDescr
 Stuff, 69

setEmail
 User, 85

setEmp_team
 Base_personnage, 18

setEmp_team_max
 User, 85

setEndu
 User, 86

setEndu_Max
 User, 86

setEsc
 Personnage, 41

setEtat
 Personnage, 41

setEtoile
 Personnage, 42
 Stuff, 70

setId_base
 Stuff, 70

setId_perso
 Personnage, 42

setId_persos
 Base_personnage, 18

setId_stuff
 Stuff, 70

setId_user
 Personnage, 43
 User, 86

setMdp
 User, 87

setNiveau
 Personnage, 43
 Stuff, 71
 User, 87

setNom
 Base_personnage, 19
 Stuff, 71

setOr_
 User, 88

setPen
 Personnage, 43

setPre
 Personnage, 44

setPseudo
 User, 88

setPv
 Base_personnage, 19

setQualite
 Base_personnage, 20
 Personnage, 44
 Stuff, 72

setSoi
 Personnage, 45

setStuff_1
 Base_personnage, 20

setStuff_2
 Base_personnage, 20

setStuff_3
 Base_personnage, 21

setStuff_4
 Base_personnage, 21

setTeam
 Personnage, 45

setTen
 Personnage, 45

setTimeEndormi
 Personnage, 46

setTypes
 Base_personnage, 21
 Personnage, 46
 Stuff, 72

setVit
 Personnage, 47

setXp
 Personnage, 47
 User, 88

setXp_Max
 User, 89

setXp_max
 Personnage, 47

soi
 Personnage, 48

- soigner
 - Personnage, 48
- Soigneur, 61
 - lancerUnSoin, 62
- Stuff, 62
 - \$atr_1, 73
 - \$atr_2, 73
 - \$cout, 73
 - \$descr, 73
 - \$etoile, 74
 - \$id_base, 74
 - \$id_stuff, 74
 - \$niveau, 74
 - \$nom, 74
 - \$qualite, 74
 - \$types, 75
 - __construct, 63
 - atr_1, 63
 - atr_2, 64
 - cout, 64
 - descr, 65
 - etoile, 65
 - hydrate, 65
 - id_base, 66
 - id_stuff, 66
 - lvlUp, 66
 - niveau, 67
 - nom, 67
 - qualite, 68
 - setAtr_1, 68
 - setAtr_2, 68
 - setCout, 69
 - setDescr, 69
 - setEtoile, 70
 - setId_base, 70
 - setId_stuff, 70
 - setNiveau, 71
 - setNom, 71
 - setQualite, 72
 - setTypes, 72
 - types, 72
- stuff_1
 - Base_personnage, 22
- stuff_2
 - Base_personnage, 22
- stuff_3
 - Base_personnage, 22
- stuff_4
 - Base_personnage, 23
- StuffManager, 75
 - __construct, 75
 - add, 76
 - count, 76
 - delete, 76
 - exists, 76
 - get, 77
 - get_base_perso, 77
 - get_niveau_sup, 77
 - getList, 77
 - getListCheat, 77
 - setDb, 77
 - sup_team, 78
 - update, 78
 - update_team, 78
- submit
 - add_perso.js, 112
 - sup_perso.js, 123
 - team.js, 124
- sup_perso.js
 - auto_load, 123
 - submit, 123
- sup_perso.php
 - \$manager, 101
 - \$perso, 101
 - chargerClasse, 101
- sup_team
 - PersonnagesManager, 60
 - StuffManager, 78
- team
 - Personnage, 49
- team.js
 - auto_load, 124
 - submit, 124
- team.php
 - cadre_team, 109
- team_ajax.php
 - \$manager, 102
 - chargerClasse, 102
- ten
 - Personnage, 49
- timeEndormi
 - Personnage, 50
- types
 - Base_personnage, 23
 - Personnage, 50
 - Stuff, 72
- update
 - PersonnagesManager, 60
 - StuffManager, 78
 - UserManager, 94
- update_team
 - PersonnagesManager, 60
 - StuffManager, 78
- User, 79
 - \$Xp_Max, 91
 - \$argent, 90
 - \$email, 90
 - \$emp_team_max, 90
 - \$endu, 90
 - \$endu_Max, 90
 - \$id_user, 90
 - \$mdp, 91
 - \$niveau, 91
 - \$or_, 91
 - \$pseudo, 91

- \$xp, [91](#)
- __construct, [79](#)
- argent, [80](#)
- email, [80](#)
- emp_team_max, [81](#)
- endu, [81](#)
- endu_Max, [81](#)
- hydrate, [82](#)
- id_user, [82](#)
- mdp, [82](#)
- niveau, [83](#)
- nomValide, [83](#)
- or_, [83](#)
- payer_argent, [84](#)
- pseudo, [84](#)
- setArgent, [84](#)
- setEmail, [85](#)
- setEmp_team_max, [85](#)
- setEndu, [86](#)
- setEndu_Max, [86](#)
- setId_user, [86](#)
- setMdp, [87](#)
- setNiveau, [87](#)
- setOr_, [88](#)
- setPseudo, [88](#)
- setXp, [88](#)
- setXp_Max, [89](#)
- xp, [89](#)
- xp_Max, [89](#)
- UserManager, [92](#)
 - __construct, [92](#)
 - add, [93](#)
 - count, [93](#)
 - delete, [93](#)
 - exists, [93](#)
 - get, [93](#)
 - getList, [93](#)
 - setDb, [94](#)
 - update, [94](#)
 - verifMdp, [94](#)
- verifMdp
 - UserManager, [94](#)
- vit
 - Personnage, [51](#)
- XML_ATTR
 - highlight.pack.js, [119](#)
- XML_COMMENT
 - highlight.pack.js, [119](#)
- XML_VALUE_APOS
 - highlight.pack.js, [119](#)
- XML_VALUE_QUOT
 - highlight.pack.js, [119](#)
- xml
 - highlight.pack.js, [119](#)
- xp
 - Personnage, [51](#)
 - User, [89](#)
- xp_Max
 - User, [89](#)
- xp_max
 - Personnage, [51](#)