

Projet e-stock

1.1

MACHON Thomas
GIMENEZ Hadrien
WAGINAIRE Nathan
GAUTHIER Robin (Tablette)

Table des matières

1	Le projet e-stock	1
1.1	Table des matières	2
2	Changelog	2
3	Installation	7
4	README	8
5	A propos	12
6	Licence GPL	12
7	Liste des choses à faire	13
8	Documentation des espaces de nommage	13
8.1	Référence de l'espace de nommage Ui	13
9	Documentation des classes	13
9.1	Référence de la classe Administrateur	13
9.1.1	Documentation des constructeurs et destructeur	14
9.1.2	Documentation des fonctions membres	14
9.1.3	Documentation des données membres	15
9.2	Référence de la classe AjoutArmoire	16
9.2.1	Description détaillée	17
9.2.2	Documentation des données membres	17
9.3	Référence de la classe AjoutUtilisateur	19
9.3.1	Description détaillée	20
9.3.2	Documentation des données membres	20
9.4	Référence de la classe Armoire	24
9.4.1	Documentation des constructeurs et destructeur	25
9.4.2	Documentation des fonctions membres	25
9.4.3	Documentation des propriétés	27
9.5	Référence de la classe Armoires	27

9.5.1	Documentation des constructeurs et destructeur	29
9.5.2	Documentation des fonctions membres	29
9.5.3	Documentation des données membres	31
9.5.4	Documentation des propriétés	31
9.6	Référence de la classe Article	32
9.6.1	Documentation des constructeurs et destructeur	32
9.6.2	Documentation des fonctions membres	33
9.6.3	Documentation des données membres	33
9.7	Référence de la classe AuthentificationBadge	34
9.7.1	Documentation des constructeurs et destructeur	36
9.7.2	Documentation des fonctions membres	36
9.7.3	Documentation des données membres	39
9.8	Référence de la classe BaseDeDonnees	40
9.8.1	Documentation des constructeurs et destructeur	41
9.8.2	Documentation des fonctions membres	42
9.8.3	Documentation des données membres	51
9.9	Référence de la classe Bouton	52
9.10	Référence de la classe Casier	53
9.10.1	Documentation des constructeurs et destructeur	54
9.10.2	Documentation des fonctions membres	55
9.10.3	Documentation des données membres	60
9.10.4	Documentation des propriétés	61
9.11	Référence de la classe Casiers	62
9.11.1	Documentation des constructeurs et destructeur	63
9.11.2	Documentation des fonctions membres	64
9.11.3	Documentation des données membres	65
9.11.4	Documentation des propriétés	66
9.12	Référence de la classe GestionBalances	67
9.12.1	Description détaillée	68
9.12.2	Documentation des constructeurs et destructeur	68
9.12.3	Documentation des fonctions membres	68

9.12.4	Documentation des données membres	70
9.13	Référence de la classe GestionCasier	71
9.13.1	Description détaillée	72
9.13.2	Documentation des données membres	72
9.14	Référence de la classe Groupe	73
9.14.1	Documentation des constructeurs et destructeur	74
9.14.2	Documentation des fonctions membres	74
9.14.3	Documentation des propriétés	75
9.15	Référence de la classe IhmAuthentification	76
9.15.1	Description détaillée	77
9.15.2	Documentation des constructeurs et destructeur	77
9.15.3	Documentation des fonctions membres	78
9.15.4	Documentation des données membres	81
9.16	Référence de la classe IhmAuthentificationSansBadge	82
9.16.1	Documentation des constructeurs et destructeur	84
9.16.2	Documentation des fonctions membres	85
9.16.3	Documentation des données membres	89
9.17	Référence de la classe IhmGestionArmoire	90
9.17.1	Documentation des constructeurs et destructeur	92
9.17.2	Documentation des fonctions membres	93
9.17.3	Documentation des données membres	100
9.18	Référence de la classe IhmGestionBalances	101
9.18.1	Documentation des données membres	102
9.19	Référence de la classe IhmGestionGroupe	102
9.19.1	Description détaillée	104
9.19.2	Documentation des constructeurs et destructeur	104
9.19.3	Documentation des fonctions membres	105
9.19.4	Documentation des données membres	110
9.20	Référence de la classe IHMPrincipale	111
9.20.1	Description détaillée	112
9.20.2	Documentation des données membres	112

9.20.3	Documentation des propriétés	115
9.21	Référence de la classe MenuArmoire	116
9.21.1	Description détaillée	117
9.21.2	Documentation des données membres	117
9.21.3	Documentation des propriétés	119
9.22	Référence de la classe ModificationUtilisateur	119
9.22.1	Description détaillée	120
9.22.2	Documentation des données membres	121
9.23	Référence de la classe Mouvement	125
9.23.1	Documentation des constructeurs et destructeur	126
9.23.2	Documentation des fonctions membres	126
9.23.3	Documentation des propriétés	130
9.24	Référence de la classe Mouvements	131
9.24.1	Documentation des constructeurs et destructeur	133
9.24.2	Documentation des fonctions membres	133
9.24.3	Documentation des données membres	134
9.24.4	Documentation des propriétés	135
9.25	Référence de la classe MouvementsArmoire	135
9.25.1	Description détaillée	136
9.25.2	Documentation des données membres	136
9.26	Référence de la classe Profil	138
9.26.1	Documentation des constructeurs et destructeur	138
9.26.2	Documentation des fonctions membres	139
9.26.3	Documentation des propriétés	140
9.27	Référence de la classe StockArmoire	141
9.27.1	Description détaillée	142
9.27.2	Documentation des fonctions membres	142
9.27.3	Documentation des données membres	142
9.27.4	Documentation des propriétés	144
9.28	Référence de la classe SuppressionUtilisateur	144
9.28.1	Description détaillée	146

9.28.2	Documentation des données membres	146
9.29	Référence de la classe Utilisateur	150
9.29.1	Documentation des constructeurs et destructeur	152
9.29.2	Documentation des fonctions membres	153
9.29.3	Documentation des données membres	158
9.30	Référence de la classe UtilisateurArmoire	160
9.30.1	Documentation des constructeurs et destructeur	161
9.30.2	Documentation des fonctions membres	162
9.30.3	Documentation des propriétés	167
9.31	Référence de la classe UtilisateursArmoire	168
9.31.1	Description détaillée	169
9.31.2	Documentation des constructeurs et destructeur	170
9.31.3	Documentation des fonctions membres	170
9.31.4	Documentation des données membres	174
9.31.5	Documentation des propriétés	176
9.32	Référence de la classe UtilisateurTerminal	177
9.32.1	Documentation des constructeurs et destructeur	179
9.32.2	Documentation des fonctions membres	179
9.32.3	Documentation des données membres	182
9.32.4	Documentation des propriétés	184
10	Documentation des fichiers	185
10.1	Référence du fichier administrateur.cpp	185
10.2	Référence du fichier administrateur.h	185
10.3	Référence du fichier AjoutArmoire.qml	185
10.3.1	Description détaillée	185
10.4	Référence du fichier AjoutUtilisateur.qml	185
10.4.1	Description détaillée	186
10.5	Référence du fichier armoire.cpp	186
10.6	Référence du fichier armoire.h	186
10.7	Référence du fichier armoires.cpp	186
10.8	Référence du fichier armoires.h	186

10.9	Référence du fichier article.cpp	186
10.10	Référence du fichier article.h	187
10.11	Référence du fichier AuthentificationBadge.cpp	187
10.12	Référence du fichier AuthentificationBadge.h	187
10.13	Référence du fichier baseDeDonnees.cpp	187
10.13.1	Description détaillée	187
10.14	Référence du fichier baseDeDonnees.cpp	187
10.14.1	Description détaillée	188
10.15	Référence du fichier baseDeDonnees.h	188
10.15.1	Description détaillée	188
10.16	Référence du fichier baseDeDonnees.h	188
10.16.1	Description détaillée	189
10.16.2	Documentation des macros	189
10.17	Référence du fichier Bouton.qml	189
10.18	Référence du fichier casier.cpp	189
10.19	Référence du fichier casier.cpp	190
10.20	Référence du fichier casier.h	190
10.21	Référence du fichier casier.h	190
10.22	Référence du fichier Casier.qml	190
10.22.1	Description détaillée	190
10.23	Référence du fichier casiers.cpp	190
10.24	Référence du fichier casiers.h	191
10.25	Référence du fichier Changelog.md	191
10.26	Référence du fichier e-stock.h	191
10.26.1	Description détaillée	191
10.26.2	Documentation des macros	191
10.27	Référence du fichier GestionBalances.cpp	192
10.28	Référence du fichier GestionBalances.h	192
10.28.1	Description détaillée	193
10.28.2	Documentation des macros	193
10.29	Référence du fichier GestionCasier.qml	193

10.29.1 Description détaillée	193
10.30 Référence du fichier groupe.cpp	193
10.31 Référence du fichier groupe.h	194
10.32 Référence du fichier IhmAuthentication.cpp	194
10.32.1 Description détaillée	194
10.33 Référence du fichier IhmAuthentication.h	194
10.33.1 Description détaillée	195
10.34 Référence du fichier IhmAuthenticationSansBadge.cpp	195
10.34.1 Description détaillée	195
10.35 Référence du fichier IhmAuthenticationSansBadge.h	195
10.36 Référence du fichier IhmGestionArmoire.cpp	196
10.36.1 Description détaillée	196
10.37 Référence du fichier IhmGestionArmoire.h	196
10.37.1 Documentation des macros	196
10.38 Référence du fichier IhmGestionBalances.cpp	197
10.39 Référence du fichier IhmGestionBalances.h	197
10.40 Référence du fichier IhmGestionGroupe.cpp	197
10.41 Référence du fichier IhmGestionGroupe.h	197
10.41.1 Description détaillée	197
10.42 Référence du fichier IHMPrincipale.qml	198
10.42.1 Description détaillée	198
10.43 Référence du fichier INSTALL.md	198
10.44 Référence du fichier main.cpp	198
10.44.1 Description détaillée	198
10.44.2 Documentation des fonctions	198
10.45 Référence du fichier main.cpp	199
10.45.1 Description détaillée	199
10.45.2 Documentation des fonctions	199
10.46 Référence du fichier MenuArmoire.qml	200
10.46.1 Description détaillée	200
10.47 Référence du fichier ModificationUtilisateur.qml	201

10.47.1 Description détaillée	201
10.48 Référence du fichier mouvement.cpp	201
10.49 Référence du fichier mouvement.h	201
10.50 Référence du fichier mouvements.cpp	201
10.51 Référence du fichier mouvements.h	201
10.52 Référence du fichier MouvementsArmoire.qml	202
10.52.1 Description détaillée	202
10.53 Référence du fichier profil.cpp	202
10.54 Référence du fichier profil.h	202
10.55 Référence du fichier README.md	202
10.56 Référence du fichier StockArmoire.qml	202
10.56.1 Description détaillée	203
10.57 Référence du fichier SuppressionUtilisateur.qml	203
10.57.1 Description détaillée	203
10.58 Référence du fichier utilisateur.cpp	203
10.59 Référence du fichier utilisateur.h	203
10.60 Référence du fichier utilisateurArmoire.cpp	203
10.61 Référence du fichier utilisateurArmoire.h	204
10.62 Référence du fichier utilisateursArmoire.cpp	204
10.63 Référence du fichier utilisateursArmoire.h	204
10.64 Référence du fichier UtilisateursArmoire.qml	204
10.64.1 Description détaillée	204
10.65 Référence du fichier utilisateurTerminal.cpp	205
10.66 Référence du fichier utilisateurTerminal.h	205
Index	207

1 Le projet e-stock

Système de gestion de stock automatisé et communicant.

Auteur

GAUTHIER Robin rgauthier2510@gmail.com
 GIMENEZ Hadrien gimenezhadrien@gmail.com
 MACHON Thomas thomaslasalle84@gmail.com
 WAGINAIRE Nathan nathanwaginaire3@gmail.com

1.1 Table des matières

- [README](#)
- [Changelog](#)
- [Installation](#)
- [Liste des choses à faire](#)
- [A propos](#)
- [Licence GPL](#)

2 Changelog

r85 | RobinG | 2019-06-04 16 :50 :01 +0200 (mar. 04 juin 2019) | 1 ligne

Ajout de l'actualisation de la page [StockArmoire](#) (timer) et création de la page [GestionCasier](#)

r84 | tvaira | 2019-06-01 18 :26 :39 +0200 (sam. 01 juin 2019) | 2 lignes

Ajout TODOs dans classe [Casier](#)

r83 | tvaira | 2019-06-01 11 :46 :10 +0200 (sam. 01 juin 2019) | 2 lignes

Mise en forme IHM

r82 | tvaira | 2019-06-01 09 :52 :20 +0200 (sam. 01 juin 2019) | 2 lignes

Pb compilation : Désactivation classe [IhmGestionBalances](#)

r81 | RobinG | 2019-05-29 18 :29 :34 +0200 (mer. 29 mai 2019) | 1 ligne

Page [StockArmoire](#) fonctionnelle : Affichage du stock en temps réel, Affichage des différents casiers présents dans l'armoire sélectionné

r80 | RobinG | 2019-05-28 18 :46 :21 +0200 (mar. 28 mai 2019) | 1 ligne

Ajout de la classe [Casiers](#), Avancement de la partie Affichage de la page [StockArmoire](#)

r79 | hgimenez | 2019-05-28 08 :19 :13 +0200 (mar. 28 mai 2019) | 2 lignes

Reception de la trame fonctionnel

r78 | RobinG | 2019-05-27 12 :19 :08 +0200 (lun. 27 mai 2019) | 1 ligne

Ajout de la classe [Casier](#)

r77 | tvaira | 2019-05-24 16 :00 :14 +0200 (ven. 24 mai 2019) | 1 ligne

Base de données v0.2

r76 | RobinG | 2019-05-23 16 :21 :37 +0200 (jeu. 23 mai 2019) | 1 ligne

Ajout des fonctionnalités : Ajout armoire et Suppression armoire

r75 | tmachon | 2019-05-23 16 :16 :48 +0200 (jeu. 23 mai 2019) | 3 lignes

Modification de [IhmGestionArmoire](#) et ajout du numéro de casier à la selectionnerArticle

r74 | tmachon | 2019-05-23 11 :57 :58 +0200 (jeu. 23 mai 2019) | 5 lignes

Ajout de la documentation Doxygen pour les classes [IhmAuthentification](#) [IhmGestionArmoire](#) et [IhmAuthentificationSansBadge](#) + rechercherArticle affiche dans la combot box la liste trouvé et le numero de casier de l'article selectionner

r73 | RobinG | 2019-05-21 17 :43 :42 +0200 (mar. 21 mai 2019) | 1 ligne

r72 | tmachon | 2019-05-21 17 :18 :17 +0200 (mar. 21 mai 2019) | 2 lignes

Ajout de la méthode rechercherArticle() dans la class [IhmGestionArmoire](#).

r71 | tmachon | 2019-05-21 08 :55 :35 +0200 (mar. 21 mai 2019) | 2 lignes

Récupération des données utilisateur complète.

r70 | tvaira | 2019-05-20 11 :12 :53 +0200 (lun. 20 mai 2019) | 1 ligne

Ajout documentation Doxygen pour tag 0.2

r69 | tvaira | 2019-05-20 11 :09 :16 +0200 (lun. 20 mai 2019) | 1 ligne

Passage des fichiers Doxygen au format Markdown

r68 | tmachon | 2019-05-17 11 :28 :05 +0200 (ven. 17 mai 2019) | 2 lignes

Authentification sans badge avec identifiant et mot de passe vérifié.

r67 | tvaira | 2019-05-13 06 :16 :24 +0200 (lun. 13 mai 2019) | 2 lignes

Vérification [IHMPricipale.qml](#)

r66 | tvaira | 2019-05-12 22 :05 :21 +0200 (dim. 12 mai 2019) | 2 lignes

Ajout gestion SQLite

r65 | tmachon | 2019-05-10 11 :12 :02 +0200 (ven. 10 mai 2019) | 1 ligne

création du tag 0.2

r64 | tvaira | 2019-05-10 06 :35 :43 +0200 (ven. 10 mai 2019) | 2 lignes

Gestion 2 BDD dans la classe [BaseDeDonnees](#)

r63 | RobinG | 2019-05-09 08 :29 :13 +0200 (jeu. 09 mai 2019) | 1 ligne

r62 | tvaira | 2019-05-08 16 :08 :57 +0200 (mer. 08 mai 2019) | 2 lignes

Vérifications démo

r61 | tvaira | 2019-05-08 11 :01 :24 +0200 (mer. 08 mai 2019) | 2 lignes

Ajout Style TableView

r60 | tvaira | 2019-05-07 19 :04 :12 +0200 (mar. 07 mai 2019) | 2 lignes

Modification pour rendre les différentes fenêtres modales

r59 | RobinG | 2019-05-06 17 :51 :14 +0200 (lun. 06 mai 2019) | 1 ligne

Ajout des fichiers [utilisateursArmoire.h](#) et [utilisateursArmoire.cpp](#)

r58 | RobinG | 2019-05-06 14 :55 :31 +0200 (lun. 06 mai 2019) | 2 lignes

Ajout navigation par page

r57 | tmachon | 2019-05-03 18 :08 :15 +0200 (ven. 03 mai 2019) | 2 lignes

Modification Authentification sans badge

r56 | RobinG | 2019-05-03 16 :32 :05 +0200 (ven. 03 mai 2019) | 1 ligne

Ajout des fichiers [utilisateursArmoire.h](#), [utilisateursArmoire.cpp](#) et [UtilisateursArmoire.qml](#), modification des fichiers [mouvements.h](#) et [mouvements.cpp](#) (Requetes SQL)

r55 | tmachon | 2019-05-03 11 :55 :44 +0200 (ven. 03 mai 2019) | 3 lignes

Ajout des méthodes : `validerAcces()`, `estAutorise()`, `estValide()`. Dans la class [IhmAuthentificationSansBadge](#).

r54 | hgimenez | 2019-05-03 11 :50 :28 +0200 (ven. 03 mai 2019) | 2 lignes

Ajout de [GestionBalances](#) parametrage du port pour recevoir les données

r53 | RobinG | 2019-05-02 18 :03 :07 +0200 (jeu. 02 mai 2019) | 1 ligne

Mise à jour des fichiers [UtilisateurArmoire.qml](#), [utilisateurArmoire.h](#) et [utilisateurArmoire.cpp](#)

r52 | RobinG | 2019-05-02 17 :41 :04 +0200 (jeu. 02 mai 2019) | 1 ligne

Ajout des fichiers [UtilisateurArmoire.qml](#), [utilisateurArmoire.h](#) et [utilisateurArmoire.cpp](#)

r51 | RobinG | 2019-05-02 17 :12 :37 +0200 (jeu. 02 mai 2019) | 1 ligne

Ajout des fichiers [utilisateurArmoire.h](#) et [utilisateurArmoire.cpp](#)

r50 | tmachon | 2019-05-02 16 :27 :57 +0200 (jeu. 02 mai 2019) | 2 lignes

Ajout de la vérification de la Date de Validite

r49 | RobinG | 2019-05-02 11 :24 :17 +0200 (jeu. 02 mai 2019) | 1 ligne

r48 | RobinG | 2019-05-02 11 :20 :20 +0200 (jeu. 02 mai 2019) | 1 ligne

r47 | RobinG | 2019-05-02 10 :59 :15 +0200 (jeu. 02 mai 2019) | 1 ligne

r46 | RobinG | 2019-05-02 10 :41 :51 +0200 (jeu. 02 mai 2019) | 2 lignes

ajout du fichier [MenuArmoire.qml](#)

r45 | tvaira | 2019-04-30 12 :21 :04 +0200 (mar. 30 avril 2019) | 1 ligne

Ajout de TODOs

r44 | tvaira | 2019-04-30 12 :11 :33 +0200 (mar. 30 avril 2019) | 1 ligne

Exemple Modele C++ et TableView

r43 | RobinG | 2019-04-25 16 :43 :41 +0200 (jeu. 25 avril 2019) | 1 ligne

Ajout des pages [MouvementArmoire](#) et modifications IHM principale

r42 | hgimenez | 2019-04-25 15 :52 :15 +0200 (jeu. 25 avril 2019) | 2 lignes

Ajout des classes [IhmGestionBalance](#)

r41 | RobinG | 2019-04-18 20 :28 :47 +0200 (jeu. 18 avril 2019) | 1 ligne

Ajout des pages [StockArmoire](#) et [MenuArmoire](#). Modification de l'IHM Principale

r40 | tvaira | 2019-04-06 10 :14 :28 +0200 (sam. 06 avril 2019) | 1 ligne

Ajout de la documentation pour le tag 0.1

r39 | RobinG | 2019-04-05 11 :50 :45 +0200 (ven. 05 avril 2019) | 1 ligne

Création du tag 0.1

r38 | RobinG | 2019-04-05 11 :46 :55 +0200 (ven. 05 avril 2019) | 1 ligne

Ajout du dossier 0.1 dans le dossier tags

r37 | RobinG | 2019-04-05 11 :45 :15 +0200 (ven. 05 avril 2019) | 1 ligne

Mise à jour Changelog

r36 | RobinG | 2019-04-05 11 :20 :46 +0200 (ven. 05 avril 2019) | 2 lignes

version 0.1

r35 | hgimenez | 2019-04-04 12 :05 :10 +0200 (jeu. 04 avril 2019) | 2 lignes

supprimerGroupe fonctionnel

r34 | tvaira | 2019-04-02 06 :56 :14 +0200 (mar. 02 avril 2019) | 2 lignes

Adresses IP des Raspberry Pi

r33 | tvaira | 2019-04-02 06 :54 :59 +0200 (mar. 02 avril 2019) | 2 lignes

Ajout classe [Casier](#) (vierge) et Clavier Florence

r32 | tvaira | 2019-04-01 10 :37 :24 +0200 (lun. 01 avril 2019) | 1 ligne

Mise à jour Doxyfile

r31 | tvaira | 2019-03-31 19 :42 :15 +0200 (dim. 31 mars 2019) | 1 ligne

Controle Qualité Revue 2

r30 | RobinG | 2019-03-29 16 :32 :40 +0100 (ven. 29 mars 2019) | 2 lignes

Ajout de la connexion à la base de données MySQL des armoires

r29 | tmachon | 2019-03-29 11 :48 :00 +0100 (ven. 29 mars 2019) | 2 lignes

Modification de la méthode estAutorise() dans la classe Badge

r28 | tvaira | 2019-03-29 06 :42 :24 +0100 (ven. 29 mars 2019) | 2 lignes

Ajout de la classe [BaseDeDonnees](#) fournie

r27 | hgimenez | 2019-03-28 16 :38 :55 +0100 (jeu. 28 mars 2019) | 2 lignes

Modification de IHMGestionGroupe ajouterGroupe() fonctionnel

r26 | tmachon | 2019-03-27 13 :42 :29 +0100 (mer. 27 mars 2019) | 3 lignes

Modification de l'[IhmAuthentificationSansBadge](#) (instanciation dynamique de l'objet [IhmAuthentificationSansBadge](#))

r25 | hgimenez | 2019-03-22 17 :36 :59 +0100 (ven. 22 mars 2019) | 2 lignes

Modification fichier .pro

r24 | hgimenez | 2019-03-22 17 :34 :39 +0100 (ven. 22 mars 2019) | 2 lignes

Ajout affichage liste des groupes

r23 | tmachon | 2019-03-22 12 :08 :48 +0100 (ven. 22 mars 2019) | 2 lignes

Ajout de la class [IhmAuthentificationSansBadge](#) squelette

r22 | hgimenez | 2019-03-22 11 :04 :22 +0100 (ven. 22 mars 2019) | 3 lignes

Ajout de la classe [IhmGestionGroupe](#) et modification dans la classe [IhmGestionArmoire](#)

r21 | tmachon | 2019-03-22 09 :27 :27 +0100 (ven. 22 mars 2019) | 2 lignes

Ajout de la class [IhmGestionArmoire](#) squelette.

r20 | tmachon | 2019-03-22 08 :44 :52 +0100 (ven. 22 mars 2019) | 2 lignes

Renommage de la classe [IhmAuthentification](#)

r19 | tmachon | 2019-03-21 15 :40 :29 +0100 (jeu. 21 mars 2019) | 1 ligne

modification de la classe IhmEstock et de la méthode validerAcces

r18 | hgimenez | 2019-03-21 14 :55 :32 +0100 (jeu. 21 mars 2019) | 1 ligne

ajout de la classe administrateur

r17 | tvaira | 2019-03-20 05 :54 :17 +0100 (mer. 20 mars 2019) | 2 lignes

Exemple d'affichage casiers

r16 | tvaira | 2019-03-17 18 :18 :44 +0100 (dim. 17 mars 2019) | 1 ligne

Terminal Mobile : exemple Boite de dialogue [AjoutArmoire](#)

r15 | tvaira | 2019-03-16 18 :11 :55 +0100 (sam. 16 mars 2019) | 2 lignes

Connexion [BaseDeDonnees](#)

r14 | tvaira | 2019-03-16 18 :09 :20 +0100 (sam. 16 mars 2019) | 2 lignes

Mise à jour de la classe [BaseDeDonnees](#)

r13 | tvaira | 2019-03-16 09 :36 :46 +0100 (sam. 16 mars 2019) | 2 lignes

Ajout de la classe [BaseDeDonnees](#)

r12 | tvaira | 2019-03-15 19 :31 :59 +0100 (ven. 15 mars 2019) | 1 ligne

Modificiation Doxyfile pour QML

r11 | hgimenez | 2019-03-15 11 :17 :53 +0100 (ven. 15 mars 2019) | 1 ligne

programme test récupération de ligne / colonne base de donnée

r10 | tvaira | 2019-03-15 05 :34 :06 +0100 (ven. 15 mars 2019) | 1 ligne

Suppression fichier inutile .pro.user

r9 | tvaira | 2019-03-15 05 :33 :32 +0100 (ven. 15 mars 2019) | 1 ligne

Initialisation de la documentation de code

r8 | RobinG | 2019-03-14 17 :00 :34 +0100 (jeu. 14 mars 2019) | 1 ligne

Ajout des pages ajouterArmoire et main.qml du terminalmobile

r7 | tmachon | 2019-03-14 16 :41 :36 +0100 (jeu. 14 mars 2019) | 1 ligne

Creation des classes : [Utilisateur](#), [Badge](#), [Article](#), [IhmStock](#)

r6 | nwaginaire | 2019-03-14 15 :05 :49 +0100 (jeu. 14 mars 2019) | 1 ligne

ajout de l'ihm

r5 | tvaira | 2019-03-09 07 :56 :18 +0100 (sam. 09 mars 2019) | 1 ligne

Initialisation Doxygen et ajout fichier SQL

r4 | RobinG | 2019-03-01 08 :33 :14 +0100 (ven. 01 mars 2019) | 1 ligne

Remplissage du fichier Changelog

r3 | RobinG | 2019-02-28 16 :37 :29 +0100 (jeu. 28 févr. 2019) | 1 ligne

Remplissage du fichier README

r2 | hgimenez | 2019-02-07 11 :30 :30 +0100 (jeu. 07 févr. 2019) | 1 ligne

ajout de TODO Changelog et README

r1 | www-data | 2019-02-06 20 :08 :27 +0100 (mer. 06 févr. 2019) | 1 ligne

Creating initial repository structure

3 Installation

Fabrication

```
$ qmake
$ make
$ sudo make install
```

Base de données

Serveur MySQL

```
$ sudo apt install mysql-server
```

e-stock

```
$ mysql -uroot -ppassword -hlocalhost < e-stock-v0.2.sql
```

Compte gestionnaire

```
$ mysql -uroot -ppassword -hlocalhost
mysql> CREATE USER 'gestionnaire'@'%' IDENTIFIED BY 'password';
mysql> GRANT ALL PRIVILEGES ON `e-stock`.* TO 'gestionnaire'@'%';
mysql> FLUSH PRIVILEGES;
```

Accès distant

```
$ sudo vim /etc/mysql/mysql.conf.d/mysqld.cnf
bind-address 0.0.0.0
```

Redémarrage

```
$ sudo systemctl restart mysql.service
```

Vérification

```
$ sudo systemctl status mysql.service
```

4 README

Nom : e-stock

Système de gestion de stock automatisé et communicant.

Numéro de version : 1.0

Auteurs

Auteur

GAUTHIER Robin rgauthier2510@gmail.com
GIMENEZ Hadrien gimenezhadrien@gmail.com
MACHON Thomas thomaslasalle84@gmail.com
WAGINAIRE Nathan nathanwaginaire3@gmail.com

Dépôt SVN

<https://svn.riouxsvn.com/e-stock>

Recette IR

- Étudiant : GAUTHIER Robin
- Étudiant : GIMENEZ Hadrien
- Étudiant : MACHON Thomas
- Étudiant : WAGINAIRE Nathan

Base de données MySQL

```

--
-- Base de données: `e-stock`
--

DROP DATABASE IF EXISTS `e-stock`;

CREATE DATABASE IF NOT EXISTS `e-stock`;

USE `e-stock`;

-----

-- Création du compte d'accès à la base de données e-stock

-- CREATE USER 'estock'@'%' IDENTIFIED BY 'password';
-- GRANT ALL PRIVILEGES ON `e-stock`.* TO 'estock'@'%';
-- FLUSH PRIVILEGES;

-----

--
-- Structure de la table `Profil`
--

CREATE TABLE IF NOT EXISTS `Profil` (
  `idProfil` int(11) NOT NULL AUTO_INCREMENT,
  `Nom` varchar(64) NOT NULL,
  PRIMARY KEY (`idProfil`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-----

INSERT INTO `Profil` (`Nom`) VALUES
('Administrateur'),
('Gestionnaire'),
('Utilisateur');

-----

--
-- Structure de la table `Groupe`
--

CREATE TABLE IF NOT EXISTS `Groupe` (
  `idGroupe` int(11) NOT NULL AUTO_INCREMENT,
  `Nom` varchar(64) NOT NULL,
  PRIMARY KEY (`idGroupe`),
  CONSTRAINT Unique_Groupe UNIQUE (`Nom`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-----

INSERT INTO `Groupe` (`Nom`) VALUES
('PROFESSEUR'),
('1-BTS-SN'),
('T-BTS-SN');

-----

--
-- Structure de la table `Utilisateur`
--

CREATE TABLE IF NOT EXISTS `Utilisateur` (
  `idUtilisateur` int(11) NOT NULL AUTO_INCREMENT,
  `idProfil` int(11) NOT NULL,
  `idGroupe` int(11) NOT NULL,
  `Nom` varchar(64) NOT NULL,
  `Prenom` varchar(64) NOT NULL,
  `DateValidite` date NOT NULL,
  `Identifiant` varchar(255) DEFAULT NULL,
  `MotDePasse` varchar(255) DEFAULT NULL,
  `Badge` varchar(11) NOT NULL,
  `Email` varchar(64) NOT NULL,
  PRIMARY KEY (`idUtilisateur`),
  CONSTRAINT Unique_Utilisateur UNIQUE (`Badge`),
  CONSTRAINT Utilisateur_fk_1 FOREIGN KEY (`idProfil`) REFERENCES Profil (`idProfil`) ON DELETE CASCADE,
  CONSTRAINT Utilisateur_fk_2 FOREIGN KEY (`idGroupe`) REFERENCES Groupe (`idGroupe`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

INSERT INTO `Utilisateur`
  (`idProfil`, `idGroupe`, `Nom`, `Prenom`, `DateValidite`, `Identifiant`, `MotDePasse`, `Badge`, `Email`)
VALUES ('1', '1', 'Vaira', 'Thierry', '2019-07-01', 'admin', '5f4dcc3b5aa765d61d8327deb882cf99', '', 'tvaira@free.fr');
INSERT INTO `Utilisateur`
  (`idProfil`, `idGroupe`, `Nom`, `Prenom`, `DateValidite`, `Identifiant`, `MotDePasse`, `Badge`, `Email`)
VALUES ('2', '1', 'Vaira', 'Thierry', '2019-07-01', 'tvaira', '5f4dcc3b5aa765d61d8327deb882cf99', '1234', 'tvaira@free.fr');
INSERT INTO `Utilisateur`
  (`idProfil`, `idGroupe`, `Nom`, `Prenom`, `DateValidite`, `Identifiant`, `MotDePasse`, `Badge`, `Email`)

```

```

VALUES('2','1','Beaumont','Jerome','2019-07-01','jbeaumont','','5678','beaumont@lasalle84.org');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','ANDREO','Michaël','2019-07-01','andreo.m','','1111','andreo.michael@outlook.fr');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','BOFFREDO','Nicolas','2019-07-01','boffredo.n','','2222','nboffredo@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','BOTELLA','Yohann','2019-07-01','botella.y','','3333','botellabroc.yohann@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','GAUTHIER','Robin','2019-07-01','gauthier.r','','5022A783','rgauthier2510@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','GIMENEZ','Hadrien','2019-07-01','gimenez.h','','02BE0261','gimenezhadrien@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','HAMMOUMA','Youssef','2019-07-01','hammouma.y','','6666','yhammouma@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','LAURAIN','Clément','2019-07-01','laurain.c','','7777','laurain.clement.contact@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','MACHON','Thomas','2019-07-01','machon.t','170796','30DDA983','thomaslasalle84@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','MELLAH','Florentin','2019-07-01','mellah.f','','9999','florentinmellah@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','REYNIER','Jacques','2019-07-01','reynier.j','','1112','reynier.jacques@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','ROSSI','Enzo','2019-07-01','rossi.e','','1113','enzo.rossi13160contact@gmail.com');
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`) VALUES('3','3','SY','Somphon',
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`) VALUES('3','3','TURLIN','Suzie
INSERT INTO `Utilisateur`
(`idProfil`,`idGroupe`,`Nom`,`Prenom`,`DateValidite`,`Identifiant`,`MotDePasse`,`Badge`,`Email`)
VALUES('3','3','WAGINAIRE','Nathan','2019-07-01','waginaire.n','','62A3F560','nathanwaginaire3@gmail.com');

-- -----
--
-- Structure de la table `Armoire`
--
CREATE TABLE IF NOT EXISTS `Armoire` (
  `idArmoire` int(11) NOT NULL AUTO_INCREMENT,
  `Nom` varchar(255) NOT NULL,
  `Description` varchar(255) DEFAULT NULL,
  `AdresseIP` varchar(15) DEFAULT NULL,
  `nbCasiers` int(11) NOT NULL DEFAULT 8,
  PRIMARY KEY (`idArmoire`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

INSERT INTO `Armoire` (`Nom`,`Description`,`AdresseIP`,`nbCasiers`) VALUES('Armoire
C01','','192.168.52.1','2');
INSERT INTO `Armoire` (`Nom`,`Description`,`AdresseIP`,`nbCasiers`) VALUES('Armoire
C02','','192.168.52.2','2');
INSERT INTO `Armoire` (`Nom`,`Description`,`AdresseIP`,`nbCasiers`) VALUES('Armoire
A01','','192.168.52.3','2');
INSERT INTO `Armoire` (`Nom`,`Description`,`AdresseIP`,`nbCasiers`) VALUES('Armoire
A02','','192.168.52.122','8');

-- -----
--
-- Structure de la table `Type`
--
CREATE TABLE IF NOT EXISTS `Type` (
  `idType` int(11) NOT NULL AUTO_INCREMENT,
  `Nom` varchar(64) NOT NULL,
  PRIMARY KEY (`idType`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- -----
INSERT INTO `Type` (`Nom`) VALUES
('Equipement'),
('Consommable');

-- -----
--
-- Structure de la table `Unite`
--
CREATE TABLE IF NOT EXISTS `Unite` (

```

```

'idUnite' int(11) NOT NULL AUTO_INCREMENT,
'Nom' varchar(64) NOT NULL,
PRIMARY KEY ('idUnite')
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

-----
INSERT INTO `Unite` (`Nom`) VALUES
('Metre'),
('Piece'),
('Pourcentage'),
('Poids g'),
('Poids kg');

```

```

--
-- Structure de la table `Comptage`
--

```

```

CREATE TABLE IF NOT EXISTS `Comptage` (
'idComptage' int(11) NOT NULL AUTO_INCREMENT,
'Nom' varchar(64) NOT NULL,
PRIMARY KEY ('idComptage')
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

-----
INSERT INTO `Comptage` (`Nom`) VALUES
('Aucun'),
('Automatique'),
('CodeBarre');

```

```

--
-- Structure de la table `Action`
--

```

```

CREATE TABLE IF NOT EXISTS `Action` (
'idAction' int(11) NOT NULL AUTO_INCREMENT,
'Nom' varchar(64) NOT NULL,
PRIMARY KEY ('idAction')
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

-----
INSERT INTO `Action` (`Nom`) VALUES
('Entree'),
('Sortie');

```

```

--
-- Structure de la table `Article`
--

```

```

CREATE TABLE IF NOT EXISTS `Article` (
'idArticle' int(11) NOT NULL AUTO_INCREMENT,
'idType' int(11) NOT NULL,
-- 'Type' enum('Equipement','Consommable'),
'Nom' varchar(255) NOT NULL,
'Code' varchar(255) NOT NULL,
'Designation' varchar(255) NOT NULL,
PRIMARY KEY ('idArticle'),
CONSTRAINT Article_fk_1 FOREIGN KEY ('idType') REFERENCES Type('idType') ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

INSERT INTO `Article` (`idType`, `Nom`, `Code`, `Designation`) VALUES('2','Vis six pans creux M2
8mm','','M2*8');
INSERT INTO `Article` (`idType`, `Nom`, `Code`, `Designation`) VALUES('2','Vis tête cylindrique M2
8mm','','M2*8');
INSERT INTO `Article` (`idType`, `Nom`, `Code`, `Designation`) VALUES('2','Vis six pans creux M2
12mm','','M2*12');
INSERT INTO `Article` (`idType`, `Nom`, `Code`, `Designation`) VALUES('2','Vis tête cylindrique M2
12mm','','M2*12');
INSERT INTO `Article` (`idType`, `Nom`, `Code`, `Designation`) VALUES('1','Fluke
i30s','2584935','Amperemetre AC/DC');
INSERT INTO `Article` (`idType`, `Nom`, `Code`, `Designation`) VALUES('1','Fluke 179','','Multimetre');

```

```

--
-- Structure de la table `Stock`
--

```

```

CREATE TABLE IF NOT EXISTS `Stock` (
'idStock' int(11) NOT NULL AUTO_INCREMENT,
'idArmoire' int(11) NOT NULL,
'idArticle' int(11) NOT NULL,
'idComptage' int(11) NOT NULL,

```

```

'idUnite' int(11) NOT NULL,
'Quantite' int(11) DEFAULT 0,
'Disponible' int(11) DEFAULT 0,
'numeroCasier' int(11) NOT NULL,
PRIMARY KEY ('idStock'),
CONSTRAINT Unique_NumeroCasier UNIQUE ('numeroCasier'),
CONSTRAINT Stock_fk_1 FOREIGN KEY ('idArmoire') REFERENCES Armoire('idArmoire') ON DELETE CASCADE,
CONSTRAINT Stock_fk_2 FOREIGN KEY ('idArticle') REFERENCES Article('idArticle') ON DELETE CASCADE,
CONSTRAINT Stock_fk_3 FOREIGN KEY ('idComptage') REFERENCES Comptage('idComptage') ON DELETE CASCADE,
CONSTRAINT Stock_fk_4 FOREIGN KEY ('idUnite') REFERENCES Unite('idUnite') ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- -----
--
-- Structure de la table 'Mouvement'
--
CREATE TABLE IF NOT EXISTS 'Mouvement' (
'idMouvement' int(11) NOT NULL AUTO_INCREMENT,
'idUtilisateur' int(11) NOT NULL,
'idStock' int(11) NOT NULL,
'idAction' int(11) NOT NULL,
-- 'Action' enum('Entree','Sortie'),
'Quantite' int(11) NOT NULL,
'Horodatage' datetime NOT NULL,
PRIMARY KEY ('idMouvement'),
CONSTRAINT Mouvement_fk_1 FOREIGN KEY ('idUtilisateur') REFERENCES Utilisateur('idUtilisateur') ON DELETE
CASCADE,
CONSTRAINT Mouvement_fk_2 FOREIGN KEY ('idStock') REFERENCES Stock('idStock') ON DELETE CASCADE,
CONSTRAINT Mouvement_fk_3 FOREIGN KEY ('idAction') REFERENCES Action('idAction') ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

-- -----

```

5 A propos

Auteur

GAUTHIER Robin rgauthier2510@gmail.com
GIMENEZ Hadrien gimenezhadrien@gmail.com
MACHON Thomas thomaslasalle84@gmail.com
WAGINAIRE Nathan nathanwaginaire3@gmail.com

Version

1.0

Date

2019

6 Licence GPL

This program is free software ; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation ; either version 2 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY ; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program ; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

7 Liste des choses à faire

Membre Casier : :fermer ()

envoyer trame fermeture

Membre Casier : :ouvrir ()

envoyer trame ouverture

Membre Utilisateur : :lire ()

Vérifier la validité de l'utilisateur

Membre Utilisateur : :lireUtilisateur (QString identifiant, QString password)

Vérifier la validité de l'utilisateur

Membre UtilisateurTerminal : :lireUtilisateur (QString p_identifiant, QString p_password, bool p_doForceLogin)

Vérifier la validité de l'utilisateur avec son mot de passe

8 Documentation des espaces de nommage

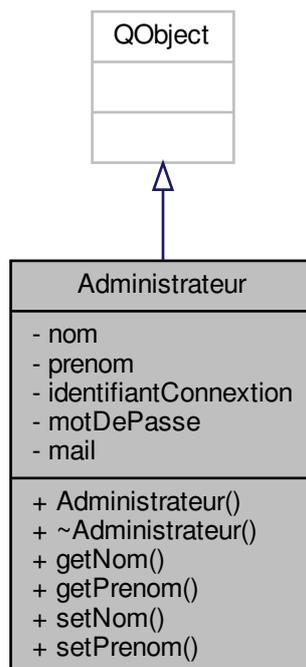
8.1 Référence de l'espace de nommage Ui

9 Documentation des classes

9.1 Référence de la classe Administrateur

```
#include <administrateur.h>
```

Graphe de collaboration de Administrateur :



Fonctions membres publiques

- [Administrateur \(\)](#)
- [~Administrateur \(\)](#)
- [char getNom \(\)](#)
- [char getPrenom \(\)](#)
- [void setNom \(\)](#)
- [void setPrenom \(\)](#)

Attributs privés

- [char nom](#)
- [char prenom](#)
- [QString identifiantConnexion](#)
- [QString motDePasse](#)
- [QString mail](#)

9.1.1 Documentation des constructeurs et destructeur

9.1.1.1 Administrateur()

```
Administrateur::Administrateur ( )
```

```
00004 {  
00005  
00006 }
```

9.1.1.2 ~Administrateur()

```
Administrateur::~~Administrateur ( )
```

```
00009 {  
00010  
00011 }
```

9.1.2 Documentation des fonctions membres

9.1.2.1 getNom()

```
char Administrateur::getNom ( )
```

Références [nom](#).

```
00014 {  
00015     return nom;  
00016 }
```

9.1.2.2 getPrenom()

```
char Administrateur::getPrenom ( )
```

Références [prenom](#).

```
00018 {  
00019     return prenom;  
00020 }
```

9.1.2.3 setNom()

```
void Administrateur::setNom ( )
```

Références [nom](#).

```
00023 {  
00024     this -> nom = nom;  
00025 }
```

9.1.2.4 setPrenom()

```
void Administrateur::setPrenom ( )
```

Références [prenom](#).

```
00028 {  
00029     this -> prenom = prenom;  
00030 }
```

9.1.3 Documentation des données membres

9.1.3.1 identifiantConnexion

```
QString Administrateur::identifiantConnexion [private]
```

9.1.3.2 mail

```
QString Administrateur::mail [private]
```

9.1.3.3 motDePasse

```
QString Administrateur::motDePasse [private]
```

9.1.3.4 nom

```
char Administrateur::nom [private]
```

Référencé par [getNom\(\)](#), et [setNom\(\)](#).

9.1.3.5 prenom

```
char Administrateur::prenom [private]
```

Référencé par [getPrenom\(\)](#), et [setPrenom\(\)](#).

La documentation de cette classe a été générée à partir des fichiers suivants :

- [administrateur.h](#)
- [administrateur.cpp](#)

9.2 Référence de la classe AjoutArmoire

La boîte de dialogue [AjoutArmoire](#).

Graphe de collaboration de AjoutArmoire :



Attributs privés

- var `modal`
- Column `edition`
Conteneur des éléments de la page qml [AjoutArmoire](#).
- Row `ligneTitre`
Conteneur du texte `labelTitre`.
- Text `labelTitre`
titre de la boîte de dialogue
- Row `ligneNomArmoire`
Conteneur des éléments `labelNomArmoire` et `saisieNomArmoire`.
- Text `labelNomArmoire`
- TextField `saisieNomArmoire`
Zone de saisie du nom de l'armoire.
- Row `ligneAdresseIP`
Conteneur des éléments `labelAdresseIP` et `saisieAdresseIP`.
- Text `labelAdresseIP`
- TextField `saisieAdresseIP`
Zone de saisie de l'adresse IP de l'armoire.
- Row `boutons`
Conteneur des boutons `boutonAjouter` et `boutonAnnuler`.
- Button `boutonAjouter`
Bouton pour ajouter une armoire.
- Button `boutonAnnuler`
Bouton pour revenir à la page [IHMPincipale](#).
- var `footer`
 pied de page

9.2.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

9.2.2 Documentation des données membres

9.2.2.1 boutonAjouter

```
Button AjoutArmoire::boutonAjouter [private]
```

9.2.2.2 boutonAnnuler

```
Button AjoutArmoire::boutonAnnuler [private]
```

9.2.2.3 boutons

```
Row AjoutArmoire::boutons [private]
```

9.2.2.4 edition

Column AjoutArmoire::edition [private]

9.2.2.5 footer

var AjoutArmoire::footer [private]

9.2.2.6 labelAdresselP

Text AjoutArmoire::labelAdresseIP [private]

9.2.2.7 labelNomArmoire

Text AjoutArmoire::labelNomArmoire [private]

9.2.2.8 labelTitre

Text AjoutArmoire::labelTitre [private]

9.2.2.9 ligneAdresselP

Row AjoutArmoire::ligneAdresseIP [private]

9.2.2.10 ligneNomArmoire

Row AjoutArmoire::ligneNomArmoire [private]

9.2.2.11 ligneTitre

Row AjoutArmoire::ligneTitre [private]

9.2.2.12 modal

var AjoutArmoire::modal [private]

9.2.2.13 saisieAdresseIP

```
TextField AjoutArmoire::saisieAdresseIP [private]
```

9.2.2.14 saisieNomArmoire

```
TextField AjoutArmoire::saisieNomArmoire [private]
```

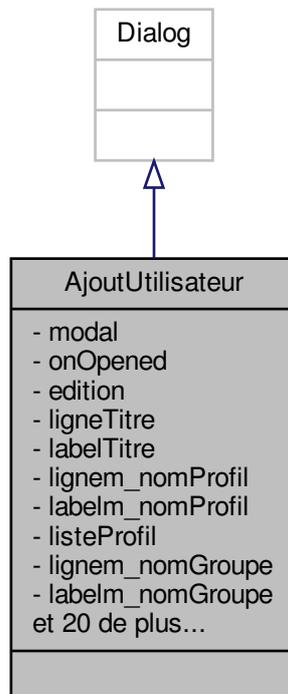
La documentation de cette classe a été générée à partir du fichier suivant :

— [AjoutArmoire.qml](#)

9.3 Référence de la classe AjoutUtilisateur

La boîte de dialogue [AjoutUtilisateur](#).

Graphe de collaboration de AjoutUtilisateur :



Attributs privés

- var `modal`
- var `onOpened`
- Column `edition`
Conteneur des éléments de la page qml [AjoutArmoire](#).
- Row `ligneTitre`
Conteneur du texte `labelTitre`.
- Text `labelTitre`
titre de la boîte de dialogue
- Row `lignem_nomProfil`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_nomProfil`
- ComboBox `listeProfil`
Liste déroulante du [Profil](#).
- Row `lignem_nomGroupe`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_nomGroupe`
- ComboBox `listeGroupe`
Zone de saisie du nom de l'`idGroupe`.
- Row `lignem_nom`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_nom`
- TextField `saisiem_nom`
Zone de saisie du nom de l'`armoire`.
- Row `lignem_prenom`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_prenom`
- TextField `saisiem_prenom`
Zone de saisie du nom de l'`armoire`.
- Row `lignem_dateDeValidite`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_dateDeValidite`
- TextField `saisiem_dateDeValidite`
Zone de saisie du nom de l'`armoire`.
- Row `lignem_identifiant`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_identifiant`
- TextField `saisiem_identifiant`
Zone de saisie du nom de l'`armoire`.
- Row `lignem_badge`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_badge`
- TextField `saisiem_badge`
Zone de saisie du nom de l'`armoire`.
- Row `boutons`
Conteneur des boutons `boutonAjouter` et `boutonAnnuler`.
- Button `boutonAjouter`
Bouton pour ajouter une armoire.
- Button `boutonAnnuler`
Bouton pour revenir à la page [IHMPricipale](#).
- var `footer`
 pied de page

9.3.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

9.3.2 Documentation des données membres

9.3.2.1 boutonAjouter

```
Button AjoutUtilisateur::boutonAjouter [private]
```

9.3.2.2 boutonAnnuler

```
Button AjoutUtilisateur::boutonAnnuler [private]
```

9.3.2.3 boutons

```
Row AjoutUtilisateur::boutons [private]
```

9.3.2.4 edition

```
Column AjoutUtilisateur::edition [private]
```

9.3.2.5 footer

```
var AjoutUtilisateur::footer [private]
```

9.3.2.6 labelm_badge

```
Text AjoutUtilisateur::labelm_badge [private]
```

9.3.2.7 labelm_dateDeValidite

```
Text AjoutUtilisateur::labelm_dateDeValidite [private]
```

9.3.2.8 labelm_identifiant

```
Text AjoutUtilisateur::labelm_identifiant [private]
```

9.3.2.9 labelm_nom

```
Text AjoutUtilisateur::labelm_nom [private]
```

9.3.2.10 labelm_nomGroupe

Text AjoutUtilisateur::labelm_nomGroupe [private]

9.3.2.11 labelm_nomProfil

Text AjoutUtilisateur::labelm_nomProfil [private]

9.3.2.12 labelm_prenom

Text AjoutUtilisateur::labelm_prenom [private]

9.3.2.13 labelTitre

Text AjoutUtilisateur::labelTitre [private]

9.3.2.14 lignem_badge

Row AjoutUtilisateur::lignem_badge [private]

9.3.2.15 lignem_dateDeValidite

Row AjoutUtilisateur::lignem_dateDeValidite [private]

9.3.2.16 lignem_identifiant

Row AjoutUtilisateur::lignem_identifiant [private]

9.3.2.17 lignem_nom

Row AjoutUtilisateur::lignem_nom [private]

9.3.2.18 lignem_nomGroupe

Row AjoutUtilisateur::lignem_nomGroupe [private]

9.3.2.19 lignem_nomProfil

Row AjoutUtilisateur::lignem_nomProfil [private]

9.3.2.20 lignem_prenom

Row AjoutUtilisateur::lignem_prenom [private]

9.3.2.21 ligneTitre

Row AjoutUtilisateur::ligneTitre [private]

9.3.2.22 listeGroupe

ComboBox AjoutUtilisateur::listeGroupe [private]

9.3.2.23 listeProfil

ComboBox AjoutUtilisateur::listeProfil [private]

9.3.2.24 modal

var AjoutUtilisateur::modal [private]

9.3.2.25 onOpened

var AjoutUtilisateur::onOpened [private]

9.3.2.26 saisiem_badge

TextField AjoutUtilisateur::saisiem_badge [private]

9.3.2.27 saisiem_dateDeValidite

TextField AjoutUtilisateur::saisiem_dateDeValidite [private]

9.3.2.28 saisie_identifiant

```
TextField AjoutUtilisateur::saisie_identifiant [private]
```

9.3.2.29 saisie_nom

```
TextField AjoutUtilisateur::saisie_nom [private]
```

9.3.2.30 saisie_prenom

```
TextField AjoutUtilisateur::saisie_prenom [private]
```

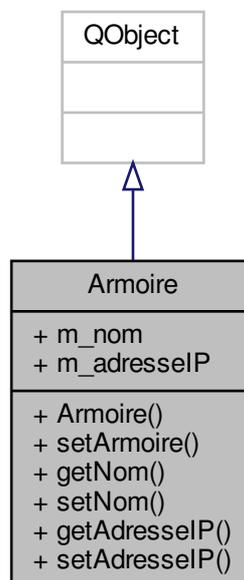
La documentation de cette classe a été générée à partir du fichier suivant :

— [AjoutUtilisateur.qml](#)

9.4 Référence de la classe Armoire

```
#include <armoire.h>
```

Graphe de collaboration de Armoire :



Signaux

- void [nomChanged](#) ()
- void [adresseIPChanged](#) ()

Fonctions membres publiques

- [Armoire](#) (QString p_nom="", QString p_adresseIP="", QObject *parent=nullptr)
- void [setArmoire](#) (QStringList p_armoire)
- QString [getNom](#) () const
- void [setNom](#) (QString p_nom)
- QString [getAdresseIP](#) () const
- void [setAdresseIP](#) (QString p_adresseIP)

Propriétés

- QString [m_nom](#)
- QString [m_adresseIP](#)

9.4.1 Documentation des constructeurs et destructeur

9.4.1.1 Armoire()

```
Armoire::Armoire (
    QString p_nom = "",
    QString p_adresseIP = "",
    QObject * parent = nullptr ) [explicit]
```

```
00003                                     : QObject (parent),
00004     m_nom(p_nom), m_adresseIP (p_adresseIP)
00005 }
```

9.4.2 Documentation des fonctions membres

9.4.2.1 adresseIPChanged

```
void Armoire::adresseIPChanged ( ) [signal]
```

9.4.2.2 getAdresseIP()

```
QString Armoire::getAdresseIP ( ) const
```

Références [m_adresseIP](#).

```
00027 {
00028     return m_adresseIP;
00029 }
```

9.4.2.3 getNom()

```
QString Armoire::getNom ( ) const
```

Références [m_nom](#).

```
00017 {  
00018     return m_nom;  
00019 }
```

9.4.2.4 nomChanged

```
void Armoire::nomChanged ( ) [signal]
```

9.4.2.5 setAdresseIP()

```
void Armoire::setAdresseIP (  
    QString p_adresseIP )
```

Références [m_adresseIP](#).

```
00032 {  
00033     m_adresseIP = p_adresseIP;  
00034 }
```

9.4.2.6 setArmoire()

```
void Armoire::setArmoire (  
    QStringList p_armoire )
```

Références [m_adresseIP](#), et [m_nom](#).

```
00008 {  
00009     if (p_armoire.size() > 0)  
00010     {  
00011         m_nom = p_armoire.at(0);  
00012         m_adresseIP = p_armoire.at(1);  
00013     }  
00014 }
```

9.4.2.7 setNom()

```
void Armoire::setNom (  
    QString p_nom )
```

Références [m_nom](#).

```
00022 {  
00023     m_nom = p_nom;  
00024 }
```

9.4.3 Documentation des propriétés

9.4.3.1 m_adresseIP

QString Armoire::m_adresseIP [read], [write]

Référencé par [getAdresseIP\(\)](#), [setAdresseIP\(\)](#), et [setArmoire\(\)](#).

9.4.3.2 m_nom

QString Armoire::m_nom [read], [write]

Référencé par [getNom\(\)](#), [setArmoire\(\)](#), et [setNom\(\)](#).

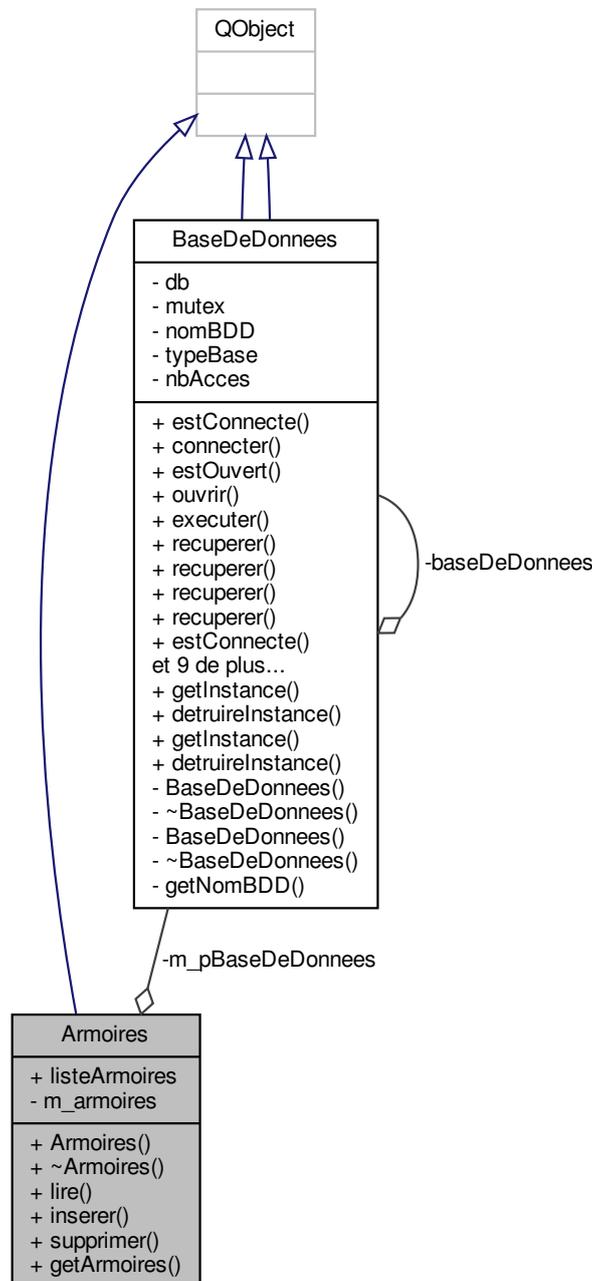
La documentation de cette classe a été générée à partir des fichiers suivants :

- [armoire.h](#)
- [armoire.cpp](#)

9.5 Référence de la classe Armoires

```
#include <armoires.h>
```

Grappe de collaboration de Armoires :



Signaux

— void [listeArmoiresChanged](#) ()

Fonctions membres publiques

— [Armoires](#) (QObject *parent=nullptr)
 — [~Armoires](#) ()
 — Q_INVOKABLE bool [lire](#) ()
 — Q_INVOKABLE bool [inserer](#) (QString p_nom, QString p_adresseIP)
 — Q_INVOKABLE bool [supprimer](#) (QString p_nom, QString p_adresseIP)
 — QVariant [getArmoires](#) ()

Propriétés

- QVariant [listeArmoires](#)

Attributs privés

- QList< QObject * > [m_armoires](#)
- [BaseDeDonnees](#) * [m_pBaseDeDonnees](#)

9.5.1 Documentation des constructeurs et destructeur

9.5.1.1 Armoires()

```
Armoires::Armoires (
    QObject * parent = nullptr ) [explicit]
```

Références [BaseDeDonnees](#) : [getInstance\(\)](#), [lire\(\)](#), et [m_pBaseDeDonnees](#).

```
00005             : QObject (p_parent)
00006 {
00007     m_pBaseDeDonnees = BaseDeDonnees::getInstance ("SQLite");
00008     lire ();
00009 }
```

9.5.1.2 ~Armoires()

```
Armoires::~Armoires ( )
```

Références [BaseDeDonnees](#) : [destruireInstance\(\)](#).

```
00012 {
00013     BaseDeDonnees::destruireInstance ("SQLite");
00014 }
```

9.5.2 Documentation des fonctions membres

9.5.2.1 getArmoires()

```
QVariant Armoires::getArmoires ( )
```

Références [m_armoires](#).

```
00069 {
00070     return QVariant::fromValue (m_armoires);
00071 }
```

9.5.2.2 inserer()

```
bool Armoires::inserer (
    QString p_nom,
    QString p_adresseIP )
```

Références [BaseDeDonnees : :executer\(\)](#), [lire\(\)](#), et [m_pBaseDeDonnees](#).

```
00039 {
00040     qDebug() << Q_FUNC_INFO << "p_nom" << p_nom << "p_adresseIP" << p_adresseIP;
00041     //requête INSERT
00042     QString requeteAjoutArmoire = "INSERT INTO Armoire (Nom, AdresseIP) VALUES ('" + p_nom + "','" +
    p_adresseIP + "')";
00043     qDebug() << Q_FUNC_INFO << "requeteAjoutArmoire" << requeteAjoutArmoire;
00044     if (m_pBaseDeDonnees->executer(requeteAjoutArmoire))
00045     {
00046         // relire la liste
00047         lire();
00048         return true;
00049     }
00050     return false;
00051 }
```

9.5.2.3 lire()

```
bool Armoires::lire ( )
```

Références [listeArmoiresChanged\(\)](#), [m_armoires](#), [m_pBaseDeDonnees](#), et [BaseDeDonnees : :recuperer\(\)](#).

Référencé par [Armoires\(\)](#), [inserer\(\)](#), et [supprimer\(\)](#).

```
00017 {
00018     QString requete = "SELECT Nom, AdresseIP FROM Armoire";
00019     QVector<QStringList> donneesArmoires;
00020
00021     if (m_pBaseDeDonnees->recuperer(requete, donneesArmoires))
00022     {
00023         QStringList donneesArmoire;
00024         m_armoires.clear();
00025         for(int i = 0; i < donneesArmoires.size(); i++)
00026         {
00027             donneesArmoire = donneesArmoires.at(i);
00028             qDebug() << Q_FUNC_INFO << "donneesArmoire" << donneesArmoire;
00029             Armoire *pArmoire = new Armoire(donneesArmoire.at(0), donneesArmoire.at(1));
00030             m_armoires.append(pArmoire);
00031         }
00032         emit listeArmoiresChanged();
00033         return true;
00034     }
00035     return false;
00036 }
```

9.5.2.4 listeArmoiresChanged

```
void Armoires::listeArmoiresChanged ( ) [signal]
```

Référencé par [lire\(\)](#).

9.5.2.5 supprimer()

```
bool Armoires::supprimer (
    QString p_nom,
    QString p_adresseIP )
```

Références [BaseDeDonnees : :executer\(\)](#), [lire\(\)](#), et [m_pBaseDeDonnees](#).

```
00054 {
00055     qDebug() << Q_FUNC_INFO << "p_nom" << p_nom << "p_adresseIP" << p_adresseIP;
00056     // Faire requête DELETE
00057     QString requeteSuppressionArmoire = "DELETE FROM Armoire WHERE Nom = '" + p_nom + "'";
00058     qDebug() << Q_FUNC_INFO << "requeteSuppressionArmoire" << requeteSuppressionArmoire;
00059     if (m_pBaseDeDonnees->executer(requeteSuppressionArmoire))
00060     {
00061         // relire la liste
00062         lire();
00063         return true;
00064     }
00065     return false;
00066 }
```

9.5.3 Documentation des données membres

9.5.3.1 m_armoires

```
QList<QObject*> Armoires::m_armoires [private]
```

Référencé par [getArmoires\(\)](#), et [lire\(\)](#).

9.5.3.2 m_pBaseDeDonnees

```
BaseDeDonnees* Armoires::m_pBaseDeDonnees [private]
```

Référencé par [Armoires\(\)](#), [inserer\(\)](#), [lire\(\)](#), et [supprimer\(\)](#).

9.5.4 Documentation des propriétés

9.5.4.1 listeArmoires

```
QVariant Armoires::listeArmoires [read]
```

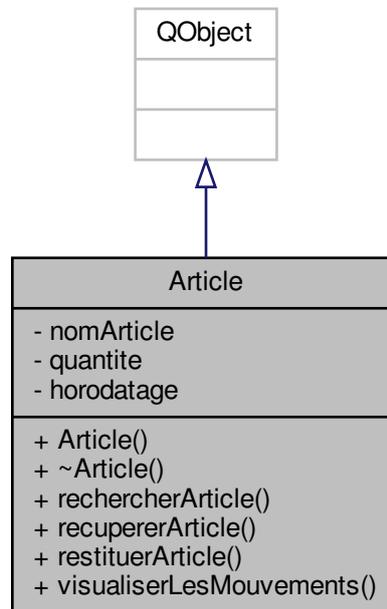
La documentation de cette classe a été générée à partir des fichiers suivants :

- [armoires.h](#)
- [armoires.cpp](#)

9.6 Référence de la classe Article

```
#include <article.h>
```

Graphe de collaboration de Article :



Fonctions membres publiques

- Article ()
- ~Article ()
- QString rechercherArticle ()
- bool recupererArticle ()
- bool restituerArticle ()
- QString visualiserLesMouvements ()

Attributs privés

- QString nomArticle
- int quantite
- QString horodatage

9.6.1 Documentation des constructeurs et destructeur

9.6.1.1 Article()

```

Article::Article ( )

00003             : nomArticle(""), quantite(0), horodatage("")
00004 {
00005
00006 }
  
```

9.6.1.2 ~Article()

```
Article::~~Article ( )
```

```
00009 {  
00010  
00011 }
```

9.6.2 Documentation des fonctions membres

9.6.2.1 rechercherArticle()

```
QString Article::rechercherArticle ( )
```

```
00014 {  
00015     return "";  
00016 }
```

9.6.2.2 recupererArticle()

```
bool Article::recupererArticle ( )
```

```
00019 {  
00020     return true;  
00021 }
```

9.6.2.3 restituerArticle()

```
bool Article::restituerArticle ( )
```

```
00024 {  
00025     return true;  
00026 }
```

9.6.2.4 visualiserLesMouvements()

```
QString Article::visualiserLesMouvements ( )
```

```
00029 {  
00030     return "";  
00031 }
```

9.6.3 Documentation des données membres

9.6.3.1 horodatage

```
QString Article::horodatage [private]
```

9.6.3.2 nomArticle

```
QString Article::nomArticle [private]
```

9.6.3.3 quantite

```
int Article::quantite [private]
```

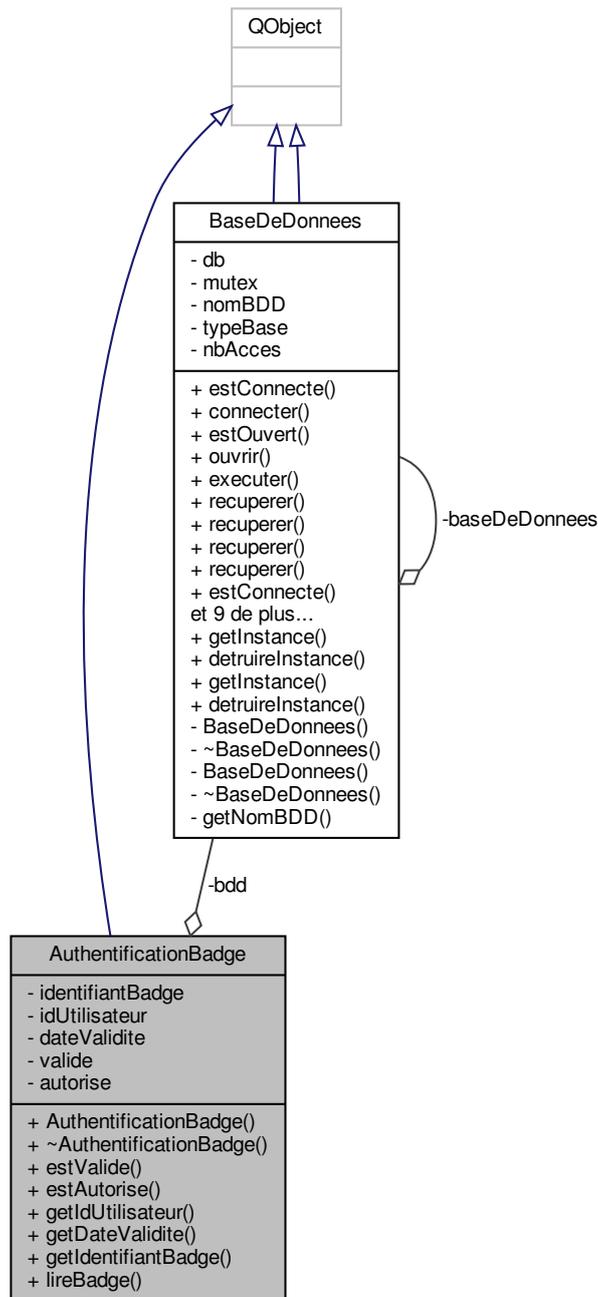
La documentation de cette classe a été générée à partir des fichiers suivants :

- [article.h](#)
- [article.cpp](#)

9.7 Référence de la classe AuthenticationBadge

```
#include <AuthenticationBadge.h>
```

Grphe de collaboration de AuthenticationBadge :



Fonctions membres publiques

- AuthenticationBadge (QObject *parent=nullptr)
- ~AuthenticationBadge ()
- bool estValide ()
- bool estAutorise ()
- QString getIdUtilisateur () const
- QString getDateValidite () const
- QString getIdentifiantBadge () const
- bool lireBadge (QString badge)

Attributs privés

- QString `identifiantBadge`
 l'UID du badge
- QString `idUtilisateur`
- QString `dateValidite`
- bool `valide`
- bool `autorise`
- `BaseDeDonnees * bdd`

9.7.1 Documentation des constructeurs et destructeur

9.7.1.1 AuthenticationBadge()

```
AuthenticationBadge::AuthenticationBadge (
    QObject * parent = nullptr )
```

Références `bdd`, `BaseDeDonnees : :connecter()`, `BaseDeDonnees : :estConnecte()`, et `BaseDeDonnees : :getInstance()`.

```
00004                                     : QObject (parent),
    valide(false), autorise(false)
00005 {
00006     qDebug() << Q_FUNC_INFO;
00007     bdd = BaseDeDonnees::getInstance();
00008     if(!bdd->estConnecte())
00009         bdd->connecter();
00010 }
```

9.7.1.2 ~AuthenticationBadge()

```
AuthenticationBadge::~AuthenticationBadge ( )
```

Références `BaseDeDonnees : :destruireInstance()`.

```
00013 {
00014     BaseDeDonnees::destruireInstance();
00015     qDebug() << Q_FUNC_INFO;
00016 }
```

9.7.2 Documentation des fonctions membres

9.7.2.1 estAutorise()

```
bool AuthenticationBadge::estAutorise ( )
```

Références `bdd`, `dateValidite`, `estValide()`, `identifiantBadge`, `idUtilisateur`, `BaseDeDonnees : :recuperer()`, et `valide`.

Référencé par `IhmAuthentication : :validerAcces()`.

```
00107 {
00108     if(!valide)
00109         return false;
00110     QString dateMaintenant = QDateTime::currentDateTime().toString("yyyy-MM-dd");
00111     QString requete = "SELECT idUtilisateur, DateValidite FROM Utilisateur WHERE Badge=' " +
    identifiantBadge + "'";
00112     QStringList donneesUtilisateur;
00113     idUtilisateur.clear();
00114     dateValidite.clear();
00115     valide = bdd->recuperer(requete, donneesUtilisateur);
00116     if (valide)
00117     {
00118         idUtilisateur = donneesUtilisateur.at(0);
00119         dateValidite = donneesUtilisateur.at(1);
00120         qDebug() << Q_FUNC_INFO << "valide" << valide << "idUtilisateur" <<
    idUtilisateur << "dateValidite" << dateValidite;
00121     }
00122     qDebug() << Q_FUNC_INFO << "valide" << valide;
00123     return (valide && estValide());
00124 }
```

9.7.2.2 estValide()

```
bool AuthenticationBadge::estValide ( )
```

Références [bdd](#), [dateValidite](#), [identifiantBadge](#), [BaseDeDonnees : :recuperer\(\)](#), et [valide](#).

Référencé par [estAutorise\(\)](#).

```
00096 {
00097     QString dateMaintenant = QDateTime::currentDateTime().toString("yyyy-MM-dd");
00098     QString requete = "SELECT DateValidite FROM Utilisateur WHERE Badge='" +
    identifiantBadge + "' AND DateValidite>='" + dateMaintenant + "'";
00099     qDebug() << Q_FUNC_INFO << "requete" << requete;
00100     QString retour;
00101     valide = bdd->recuperer(requete, retour);
00102     qDebug() << Q_FUNC_INFO << "valide" << valide << "dateValidite" <<
    dateValidite;
00103     return valide;
00104 }
```

9.7.2.3 getDateValidite()

```
QString AuthenticationBadge::getDateValidite ( ) const
```

Références [dateValidite](#).

```
00132 {
00133     return dateValidite;
00134 }
```

9.7.2.4 getIdentifiantBadge()

```
QString AuthenticationBadge::getIdentifiantBadge ( ) const
```

Références [identifiantBadge](#).

```
00137 {
00138     return identifiantBadge;
00139 }
```

9.7.2.5 getIdUtilisateur()

```
QString AuthenticationBadge::getIdUtilisateur ( ) const
```

Références [idUtilisateur](#).

Référencé par [IhmAuthentication : :validerAcces\(\)](#).

```
00127 {
00128     return idUtilisateur;
00129 }
```

9.7.2.6 lireBadge()

```
bool AuthentificationBadge::lireBadge (
    QString badge )
```

Références [identifiantBadge](#), et [valide](#).

Référencé par [IhmAuthentification](#) : [:validerAcces\(\)](#).

```
00019 {
00020     //badge = "RFIDMàBDD'èCD"; // test
00021     if(badge.isEmpty())
00022         return false;
00023
00024     qDebug() << Q_FUNC_INFO << "badge" << badge;
00025
00026     QString messageNettoye;
00027
00028     if(!badge.isEmpty())
00029     {
00030         // les informations lues proviennent d'un clavier US en QWERTY
00031         // remplacement caractères QWERTY
00032         messageNettoye = badge.replace(QString::fromUtf8("Q"), "A");
00033         messageNettoye = badge.replace(QString::fromUtf8("W"), "Z");
00034         messageNettoye = badge.replace(QString::fromUtf8("q"), "a");
00035         messageNettoye = badge.replace(QString::fromUtf8("w"), "z");
00036         messageNettoye = badge.replace(QString::fromUtf8("M"), ":");
00037         // remplacement caractères US
00038         messageNettoye = badge.replace(QString::fromUtf8("à"), "0");
00039         messageNettoye = badge.replace(QString::fromUtf8("&"), "1");
00040         messageNettoye = badge.replace(QString::fromUtf8("é"), "2");
00041         messageNettoye = badge.replace(QString::fromUtf8("\\"), "3");
00042         messageNettoye = badge.replace(QString::fromUtf8("'"), "4");
00043         messageNettoye = badge.replace(QString::fromUtf8("("), "5");
00044         messageNettoye = badge.replace(QString::fromUtf8("-"), "6");
00045         messageNettoye = badge.replace(QString::fromUtf8("è"), "7");
00046         messageNettoye = badge.replace(QString::fromUtf8("_"), "8");
00047         messageNettoye = badge.replace(QString::fromUtf8("ç"), "9");
00048         //qDebug() << Q_FUNC_INFO << messageNettoye;
00049
00050         // les informations du badge doivent commencer par "RFID:"
00051         if(messageNettoye.startsWith("RFID:"))
00052         {
00053             QStringList champs = messageNettoye.split(":");
00054             //qDebug() << Q_FUNC_INFO << champs;
00055
00056             if(champs.count() == 2 && champs.at(0) == "RFID")
00057             {
00058                 // récupération de l'UID du badge
00059                 QString UID = champs.at(1);
00060
00061                 // vérification de l'UID du badge (Un UID est une valeur hexadécimale valide)
00062                 bool ok;
00063                 qulonglong uid = UID.toULongLong(&ok, 16);
00064                 //qDebug() << Q_FUNC_INFO << UID << uid << ok;
00065                 if(ok)
00066                 {
00067                     //labelUID->setText("UID : 0x" + UID + "\nUID : " + QString::number(uid));
00068                     identifiantBadge = UID;
00069                     qDebug() << Q_FUNC_INFO << "identifiantBadge" <<
00070                     identifiantBadge;
00071                     valide = true;
00072                     return true;
00073                 }
00074                 else
00075                 {
00076                     valide = false;
00077                 }
00078             }
00079             else
00080             {
00081                 valide = false;
00082             }
00083         }
00084         else
00085         {
00086             valide = false;
00087         }
00088     }
00089     else
00090     {
00091         valide = false;
00092     }
00093     return false;
00093 }
```

9.7.3 Documentation des données membres

9.7.3.1 autorise

```
bool AuthenticationBadge::autorise [private]
```

9.7.3.2 bdd

```
BaseDeDonnees* AuthenticationBadge::bdd [private]
```

Référencé par [AuthenticationBadge\(\)](#), [estAutorise\(\)](#), et [estValide\(\)](#).

9.7.3.3 dateValidite

```
QString AuthenticationBadge::dateValidite [private]
```

Référencé par [estAutorise\(\)](#), [estValide\(\)](#), et [getDateValidite\(\)](#).

9.7.3.4 identifiantBadge

```
QString AuthenticationBadge::identifiantBadge [private]
```

Référencé par [estAutorise\(\)](#), [estValide\(\)](#), [getIdentifiantBadge\(\)](#), et [lireBadge\(\)](#).

9.7.3.5 idUtilisateur

```
QString AuthenticationBadge::idUtilisateur [private]
```

Référencé par [estAutorise\(\)](#), et [getIdUtilisateur\(\)](#).

9.7.3.6 valide

```
bool AuthenticationBadge::valide [private]
```

Référencé par [estAutorise\(\)](#), [estValide\(\)](#), et [lireBadge\(\)](#).

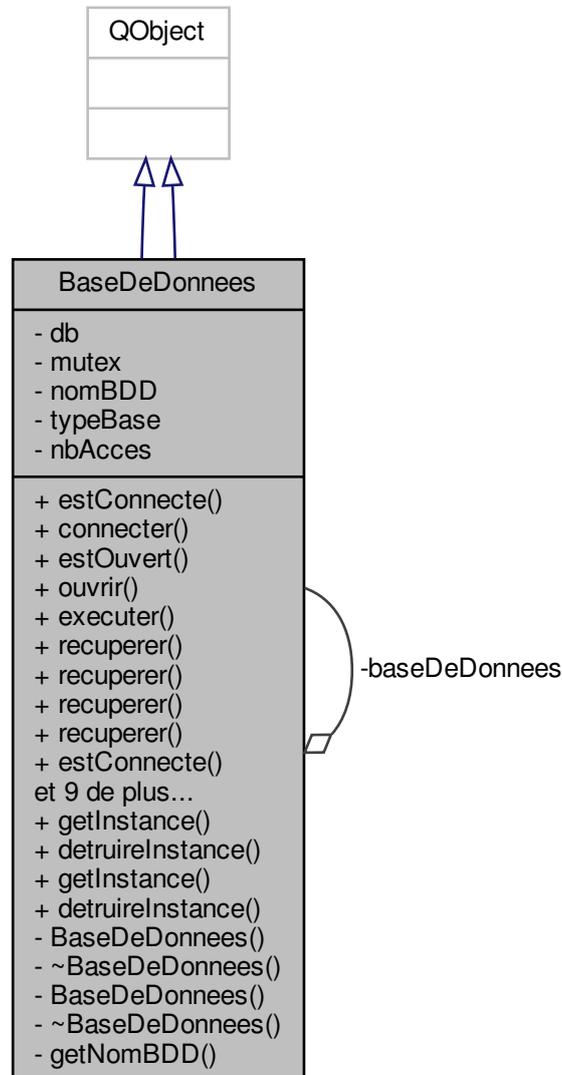
La documentation de cette classe a été générée à partir des fichiers suivants :

- [AuthenticationBadge.h](#)
- [AuthenticationBadge.cpp](#)

9.8 Référence de la classe BaseDeDonnees

```
#include <baseDeDonnees.h>
```

Graphe de collaboration de BaseDeDonnees :



Fonctions membres publiques

- bool `estConnecte` ()
- bool `connecter` (QString nomBase=`BDD_NOMBASE`, QString username=`BDD_USERNAME`, QString password=`BDD_PAS`←
`SWORD`, QString serveur=`BDD_SERVEUR`)
- bool `estOuvert` ()
- bool `ouvrir` (QString fichierBase)
- bool `executer` (QString requete)
- bool `recuperer` (QString requete, QString &donnees)
- bool `recuperer` (QString requete, QStringList &donnees)
- bool `recuperer` (QString requete, QVector< QString > &donnees)
- bool `recuperer` (QString requete, QVector< QStringList > &donnees)
- Q_INVOKABLE bool `estConnecte` ()

- Q_INVOKABLE void [deconnecter](#) ()
- Q_INVOKABLE bool [connecter](#) (QString nomBase, QString username, QString password, QString serveur)
- Q_INVOKABLE bool [estOuvert](#) ()
- Q_INVOKABLE bool [ouvrir](#) (QString fichierBase)
- Q_INVOKABLE bool [executer](#) (QString requete)
- Q_INVOKABLE bool [recuperer](#) (QString requete, QString &donnees)
- Q_INVOKABLE bool [recuperer](#) (QString requete, QStringList &donnees)
- Q_INVOKABLE bool [recuperer](#) (QString requete, QVector< QString > &donnees)
- Q_INVOKABLE bool [recuperer](#) (QString requete, QVector< QStringList > &donnees)

Fonctions membres publiques statiques

- static [BaseDeDonnees](#) * [getInstance](#) (QString type="QMYSQL")
- static void [decrireInstance](#) ()
- static [BaseDeDonnees](#) * [getInstance](#) (QString type)
- static void [decrireInstance](#) (QString type)

Fonctions membres privées

- [BaseDeDonnees](#) (QString type)
- ~[BaseDeDonnees](#) ()
- [BaseDeDonnees](#) (QString type)
- ~[BaseDeDonnees](#) ()
- QSqlDatabase [getNomBDD](#) ()

Attributs privés

- QSqlDatabase [db](#)
- QMutex [mutex](#)
- QString [nomBDD](#)

Attributs privés statiques

- static [BaseDeDonnees](#) * [baseDeDonnees](#) = NULL
- static QString [typeBase](#) = "QMYSQL"
- static int [nbAcces](#) = 0

9.8.1 Documentation des constructeurs et destructeur

9.8.1.1 BaseDeDonnees() [1/2]

```
BaseDeDonnees::BaseDeDonnees (
    QString type ) [private]
```

Références [db](#), et [typeBase](#).

Référencé par [getInstance\(\)](#).

```
00023 {
00024     #ifdef DEBUG_BASEDEDONNEES
00025     qDebug() << Q_FUNC_INFO << type;
00026     #endif
00027     db = QSqlDatabase::addDatabase(type);
00028     typeBase = type;
00029 }
```

9.8.1.2 ~BaseDeDonnees() [1/2]

```
BaseDeDonnees::~~BaseDeDonnees ( ) [private]
```

```
00032 {
00033     #ifdef DEBUG_BASEDEDONNEES
00034     qDebug() << Q_FUNC_INFO;
00035     #endif
00036 }
```

9.8.1.3 BaseDeDonnees() [2/2]

```
BaseDeDonnees::BaseDeDonnees (
    QString type ) [private]
```

9.8.1.4 ~BaseDeDonnees() [2/2]

```
BaseDeDonnees::~~BaseDeDonnees ( ) [private]
```

9.8.2 Documentation des fonctions membres

9.8.2.1 connecter() [1/2]

```
bool BaseDeDonnees::connecter (
    QString nomBase = BDD_NOMBASE,
    QString username = BDD_USERNAME,
    QString password = BDD_PASSWORD,
    QString serveur = BDD_SERVEUR )
```

Références [db](#), [mutex](#), et [typeBase](#).

Référencé par [AuthentificationBadge : :AuthentificationBadge\(\)](#), [deconnecter\(\)](#), [IhmAuthentification : :IhmAuthentification\(\)](#), [IhmAuthentificationSansBadge : :IhmAuthentificationSansBadge\(\)](#), [IhmGestionArmoire : :IhmGestionArmoire\(\)](#), [IhmGestionGroupe : :IhmGestionGroupe\(\)](#), et [Utilisateur : :Utilisateur\(\)](#).

```
00077 {
00078     if(typeBase != "MYSQL")
00079         return false;
00080     QMutexLocker verrou(&mutex);
00081     if(!db.isOpen())
00082     {
00083         db.setHostName(serveur);
00084         db.setUserName(username);
00085         db.setPassword(password);
00086         db.setDatabaseName(nomBase);
00087
00088         #ifdef DEBUG_BASEDEDONNEES
00089         qDebug() << Q_FUNC_INFO;
00090         qDebug() << "HostName : " << db.hostName();
00091         qDebug() << "UserName : " << db.userName();
00092         qDebug() << "DatabaseName : " << db.databaseName();
00093         #endif
00094         if(db.open())
00095         {
00096             #ifdef DEBUG_BASEDEDONNEES
00097             qDebug() << Q_FUNC_INFO << QString::fromUtf8("Connexion réussie à %1").arg(
00098                 db.hostName());
00099             #endif
00100         }
00101     }
00102 }
```

```

00099         return true;
00100     }
00101     else
00102     {
00103         qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : impossible de se connecter à la base de
données !");
00104         #ifdef DEBUG_BASEDEDONNEES
00105         QMessageBox::critical(0, QString::fromUtf8("e-stock"), QString::fromUtf8("Impossible de se
connecter à la base de données !"));
00106         #endif
00107         return false;
00108     }
00109 }
00110 else
00111     return true;
00112 }

```

9.8.2.2 connecter() [2/2]

```

Q_INVOKABLE bool BaseDeDonnees::connecter (
    QString nomBase,
    QString username,
    QString password,
    QString serveur )

```

9.8.2.3 deconnecter()

```
void BaseDeDonnees::deconnecter ( )
```

Références [connecter\(\)](#), [db](#), [estOuvert\(\)](#), [executer\(\)](#), [getNomBDD\(\)](#), [mutex](#), [ouvrir\(\)](#), et [recuperer\(\)](#).

Référencé par [destruireInstance\(\)](#).

```

00139 {
00140     QMutexLocker verrou(&mutex);
00141     if(getNomBDD().isOpen())
00142     {
00143         qDebug() << Q_FUNC_INFO << getNomBDD().driverName();
00144         getNomBDD().close();
00145     }
00146 }

```

9.8.2.4 detruireInstance() [1/2]

```
void BaseDeDonnees::detruireInstance ( ) [static]
```

Références [baseDeDonnees](#), et [nbAcces](#).

Référencé par [main\(\)](#), [Armoires : :~Armoires\(\)](#), [AuthentificationBadge : :~AuthentificationBadge\(\)](#), [Casiers : :~Casiers\(\)](#), [IhmAuthentification : :~IhmAuthentification\(\)](#), [IhmAuthentificationSansBadge : :~IhmAuthentificationSansBadge\(\)](#), [IhmGestionArmoire : :~IhmGestionArmoire\(\)](#), [IhmGestionGroupe : :~IhmGestionGroupe\(\)](#), [Mouvements : :~Mouvements\(\)](#), [Utilisateur : :~Utilisateur\(\)](#), [UtilisateursArmoire : :~UtilisateursArmoire\(\)](#), et [UtilisateurTerminal : :~UtilisateurTerminal\(\)](#).

```

00052 {
00053     // instance ?
00054     if(baseDeDonnees != NULL)
00055     {
00056         if(nbAcces > 0)
00057             nbAcces--;
00058         #ifdef DEBUG_BASEDEDONNEES
00059         qDebug() << Q_FUNC_INFO << "nbAcces restants" << nbAcces;
00060         #endif
00061         // dernier ?
00062         if(nbAcces == 0)
00063         {
00064             delete baseDeDonnees;
00065             baseDeDonnees = NULL;
00066         }
00067     }
00068 }

```

9.8.2.5 detruireInstance() [2/2]

```
void BaseDeDonnees::detruireInstance (
    QString type ) [static]
```

Références [baseDeDonnees](#), [BDD_MYSQL](#), [BDD_SQLITE](#), [deconnecter\(\)](#), et [nbAcces](#).

```
00082 {
00083     if(type == "QSQLITE")
00084     {
00085         if (baseDeDonnees[BDD_SQLITE] != NULL)
00086         {
00087             if (nbAcces[BDD_SQLITE] > 0)
00088             {
00089                 nbAcces[BDD_SQLITE]--;
00090             }
00091         }
00092         #ifdef DEBUG_BASEDEDONNEES
00093         qDebug() << Q_FUNC_INFO << "nbAcces" << nbAcces[BDD_SQLITE] << type;
00094         #endif
00095         if (nbAcces[BDD_SQLITE] == 0)
00096         {
00097             baseDeDonnees[BDD_SQLITE]->deconnecter();
00098             delete baseDeDonnees[BDD_SQLITE];
00099             baseDeDonnees[BDD_SQLITE] = NULL;
00100         }
00101     }
00102     else if(type == "QMYSQL")
00103     {
00104         if (baseDeDonnees[BDD_MYSQL] != NULL)
00105         {
00106             if (nbAcces[BDD_MYSQL] > 0)
00107             {
00108                 nbAcces[BDD_MYSQL]--;
00109             }
00110         }
00111         #ifdef DEBUG_BASEDEDONNEES
00112         qDebug() << Q_FUNC_INFO << "nbAcces" << nbAcces[BDD_MYSQL] << type;
00113         #endif
00114         if (nbAcces[BDD_MYSQL] == 0)
00115         {
00116             baseDeDonnees[BDD_MYSQL]->deconnecter();
00117             delete baseDeDonnees[BDD_MYSQL];
00118             baseDeDonnees[BDD_MYSQL] = NULL;
00119         }
00120     }
00121     else
00122     {
00123         qDebug() << Q_FUNC_INFO << "type" << type << "non pris en charge";
00124     }
00125 }
```

9.8.2.6 estConnecte() [1/2]

```
bool BaseDeDonnees::estConnecte ( )
```

Références [db](#), et [mutex](#).

Référencé par [AuthentificationBadge](#) : [:AuthentificationBadge\(\)](#), [getNomBDD\(\)](#), [IhmAuthentification](#) : [:IhmAuthentification\(\)](#), [IhmAuthentificationSansBadge](#) : [:IhmAuthentificationSansBadge\(\)](#), [IhmGestionArmoire](#) : [:IhmGestionArmoire\(\)](#), [IhmGestionGroupe](#) : [:IhmGestionGroupe\(\)](#), et [Utilisateur](#) : [:Utilisateur\(\)](#).

```
00071 {
00072     QMutexLocker verrou(&mutex);
00073     return db.isOpen();
00074 }
```

9.8.2.7 estConnecte() [2/2]

```
Q_INVOKABLE bool BaseDeDonnees::estConnecte ( )
```

9.8.2.8 estOuvert() [1/2]

```
bool BaseDeDonnees::estOuvert ( )
```

Références [db](#), et [mutex](#).

Référencé par [deconnecter\(\)](#).

```
00115 {
00116     QMutexLocker verrou(&mutex);
00117     return db.isOpen();
00118 }
```

9.8.2.9 estOuvert() [2/2]

```
Q_INVOKABLE bool BaseDeDonnees::estOuvert ( )
```

9.8.2.10 executer() [1/2]

```
bool BaseDeDonnees::executer (
    QString requete )
```

Références [db](#), et [mutex](#).

Référencé par [IhmGestionGroupe : :ajouterGroupe\(\)](#), [deconnecter\(\)](#), [Armoires : :inserer\(\)](#), [UtilisateursArmoire : :inserer\(\)](#), [UtilisateursArmoire : :modifier\(\)](#), [IhmGestionGroupe : :modifierGroupe\(\)](#), [Armoires : :supprimer\(\)](#), [UtilisateursArmoire : :supprimer\(\)](#), et [IhmGestionGroupe : :supprimerGroupe\(\)](#).

```
00153 {
00154     QMutexLocker verrou(&mutex);
00155     QSqlQuery r;
00156     bool retour;
00157
00158     if(db.isOpen())
00159     {
00160         if(requete.contains("UPDATE") || requete.contains("INSERT") || requete.contains("DELETE"))
00161         {
00162             retour = r.exec(requete);
00163             #ifdef DEBUG_BASEDEDONNEES
00164             qDebug() << Q_FUNC_INFO << QString::fromUtf8("Retour %1 pour la requete : %2").arg(
00165                 QString::number(retour)).arg(requete);
00166             #endif
00167             if(retour)
00168             {
00169                 return true;
00170             }
00171             else
00172             {
00173                 qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : %1 pour la requête %2").arg(r.
00174                     lastError().text()).arg(requete);
00175                 return false;
00176             }
00177             else
00178             {
00179                 qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : requête %1 non autorisée !").arg(requete);
00180                 return false;
00181             }
00182             else
00183             return false;
00184
00185 }
```

9.8.2.11 `executer()` [2/2]

```
Q_INVOKABLE bool BaseDeDonnees::executer (
    QString requete )
```

9.8.2.12 `getInstance()` [1/2]

```
BaseDeDonnees * BaseDeDonnees::getInstance (
    QString type = "QMYSQL" ) [static]
```

Références [BaseDeDonnees\(\)](#), [baseDeDonnees](#), et [nbAcces](#).

Référencé par [Armoires : :Armoires\(\)](#), [AuthentificationBadge : :AuthentificationBadge\(\)](#), [Casiers : :Casiers\(\)](#), [IhmAuthentification : :<-IhmAuthentification\(\)](#), [IhmAuthentificationSansBadge : :IhmAuthentificationSansBadge\(\)](#), [IhmGestionArmoire : :IhmGestionArmoire\(\)](#), [IhmGestionGroupe : :IhmGestionGroupe\(\)](#), [main\(\)](#), [Mouvements : :Mouvements\(\)](#), [Utilisateur : :Utilisateur\(\)](#), [UtilisateursArmoire : :<-UtilisateursArmoire\(\)](#), et [UtilisateurTerminal : :UtilisateurTerminal\(\)](#).

```
00039 {
00040     if (baseDeDonnees == NULL)
00041         baseDeDonnees = new BaseDeDonnees (type);
00042
00043     nbAcces++;
00044     #ifdef DEBUG_BASEDEDONNEES
00045     qDebug() << Q_FUNC_INFO << "nbAcces" << nbAcces;
00046     #endif
00047
00048     return baseDeDonnees;
00049 }
```

9.8.2.13 `getInstance()` [2/2]

```
static BaseDeDonnees* BaseDeDonnees::getInstance (
    QString type ) [static]
```

9.8.2.14 `getNomBDD()`

```
QSqlDatabase BaseDeDonnees::getNomBDD ( ) [private]
```

Références [estConnecte\(\)](#), [mutex](#), et [nomBDD](#).

Référencé par [deconnecter\(\)](#).

```
00128 {
00129     return QSqlDatabase::database (nomBDD, false);
00130 }
```

9.8.2.15 ouvrir() [1/2]

```
bool BaseDeDonnees::ouvrir (
    QString fichierBase )
```

Références [db](#), [mutex](#), et [typeBase](#).

Référencé par [deconnecter\(\)](#), et [main\(\)](#).

```
00121 {
00122     if(typeBase != "QSQLITE")
00123         return false;
00124     QMutexLocker verrou(&mutex);
00125     if(!db.isOpen())
00126     {
00127         db.setDatabaseName(fichierBase);
00128
00129         #ifdef DEBUG_BASEDEDONNEES
00130         qDebug() << Q_FUNC_INFO << db.databaseName();
00131         #endif
00132         if(db.open())
00133         {
00134             #ifdef DEBUG_BASEDEDONNEES
00135             qDebug() << Q_FUNC_INFO << QString::fromUtf8("Ouvertir réussie à %1").arg(
db.databaseName());
00136             #endif
00137
00138             return true;
00139         }
00140         else
00141         {
00142             qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : impossible d'ouvrir la base de données !"
);
00143             QMessageBox::critical(0, QString::fromUtf8("BaseDeDonnees"), QString::fromUtf8("Impossible
d'ouvrir la base de données !"));
00144             return false;
00145         }
00146     }
00147     else
00148         return true;
00149 }
```

9.8.2.16 ouvrir() [2/2]

```
Q_INVOKABLE bool BaseDeDonnees::ouvrir (
    QString fichierBase )
```

9.8.2.17 recuperer() [1/8]

```
bool BaseDeDonnees::recuperer (
    QString requete,
    QString & donnees )
```

Références [db](#), et [mutex](#).

Référencé par [IhmGestionArmoire : :afficherCasiers\(\)](#), [deconnecter\(\)](#), [IhmAuthentificationSansBadge : :estAutorise\(\)](#), [AuthentificationBadge : :estAutorise\(\)](#), [IhmAuthentificationSansBadge : :estValide\(\)](#), [AuthentificationBadge : :estValide\(\)](#), [Armoires : :lire\(\)](#), [Mouvements : :lire\(\)](#), [UtilisateursArmoire : :lire\(\)](#), [Casiers : :lire\(\)](#), [Utilisateur : :lire\(\)](#), [UtilisateursArmoire : :lireGroupe\(\)](#), [UtilisateursArmoire : :lireProfil\(\)](#), [UtilisateurTerminal : :lireUtilisateur\(\)](#), [Utilisateur : :lireUtilisateur\(\)](#), [IhmGestionGroupe : :listerGroupe\(\)](#), [IhmGestionArmoire : :rechercherArticle\(\)](#), [IhmGestionArmoire : :selectionnerCasier\(\)](#), et [IhmGestionGroupe : :selectionnerGroupe\(\)](#).

```

00191 {
00192     QMutexLocker verrou(&mutex);
00193     QSqlQuery r;
00194     bool retour;
00195
00196     if(db.isOpen())
00197     {
00198         if(requete.contains("SELECT"))
00199         {
00200             retour = r.exec(requete);
00201             #ifdef DEBUG_BASEDEDONNEES
00202             qDebug() << Q_FUNC_INFO << QString::fromUtf8("Retour %1 pour la requete : %2").arg(
00203             QString::number(retour)).arg(requete);
00204             #endif
00205             if(retour)
00206             {
00207                 // on se positionne sur l'enregistrement
00208                 r.first();
00209
00210                 // on vérifie l'état de l'enregistrement retourné
00211                 if(!r.isValid())
00212                 {
00213                     #ifdef DEBUG_BASEDEDONNEES
00214                     qDebug() << Q_FUNC_INFO << QString::fromUtf8("Résultat non valide !");
00215                     #endif
00216                     return false;
00217                 }
00218
00219                 // on récupère sous forme de QString la valeur du champ
00220                 if(r.isNull(0))
00221                 {
00222                     #ifdef DEBUG_BASEDEDONNEES
00223                     qDebug() << Q_FUNC_INFO << QString::fromUtf8("Aucun résultat !");
00224                     #endif
00225                     return false;
00226                 }
00227                 donnees = r.value(0).toString();
00228                 #ifdef DEBUG_BASEDEDONNEES
00229                 qDebug() << Q_FUNC_INFO << "Enregistrement -> " << donnees;
00230                 #endif
00231                 return true;
00232             }
00233             else
00234             {
00235                 qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : %1 pour la requête %2").arg(r.
00236                 lastError().text()).arg(requete);
00237                 return false;
00238             }
00239             else
00240             {
00241                 qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : requête %1 non autorisée !").arg(requete
00242                 );
00243                 return false;
00244             }
00245             else
00246             return false;
00247         }
00248     }
00249 }

```

9.8.2.18 recuperer() [2/8]

```

bool BaseDeDonnees::recuperer (
    QString requete,
    QStringList & donnees )

```

Références `db`, et `mutex`.

```

00252 {
00253     QMutexLocker verrou(&mutex);
00254     QSqlQuery r;
00255     bool retour;
00256
00257     if(db.isOpen())
00258     {
00259         if(requete.contains("SELECT"))
00260         {
00261             retour = r.exec(requete);
00262             #ifdef DEBUG_BASEDEDONNEES
00263             qDebug() << Q_FUNC_INFO << "<BaseDeDonnees::recuperer(QString, QStringList)> retour %1 pour

```

```

    la requete : %2").arg(QString::number(retour)).arg(requete);
00264     #endif
00265     if(retour)
00266     {
00267         // on se positionne sur l'enregistrement
00268         r.first();
00269
00270         // on vérifie l'état de l'enregistrement retourné
00271         if(!r.isValid())
00272         {
00273             #ifdef DEBUG_BASEDEDONNEES
00274             qDebug() << Q_FUNC_INFO << QString::fromUtf8("Résultat non valide !");
00275             #endif
00276             return false;
00277         }
00278
00279         // on récupère sous forme de QString la valeur de tous les champs sélectionnés
00280         // et on les stocke dans une liste de QString
00281         for(int i=0;i<r.record().count();i++)
00282             if(!r.isNull(i))
00283                 donnees << r.value(i).toString();
00284         #ifdef DEBUG_BASEDEDONNEES
00285         qDebug() << Q_FUNC_INFO << "Enregistrement -> " << donnees;
00286         #endif
00287         return true;
00288     }
00289     else
00290     {
00291         qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : %1 pour la requête %2").arg(r.
lastError().text()).arg(requete);
00292         return false;
00293     }
00294 }
00295 else
00296 {
00297     qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : requête %1 non autorisée !").arg(requete
);
00298     return false;
00299 }
00300 }
00301 else
00302     return false;
00303 }

```

9.8.2.19 recuperer() [3/8]

```

bool BaseDeDonnees::recuperer (
    QString requete,
    QVector< QString > & donnees )

```

Références [db](#), et [mutex](#).

```

00309 {
00310     QMutexLocker verrou(&mutex);
00311     QSqlQuery r;
00312     bool retour;
00313     QString data;
00314
00315     if(db.isOpen())
00316     {
00317         if(requete.contains("SELECT"))
00318         {
00319             retour = r.exec(requete);
00320             #ifdef DEBUG_BASEDEDONNEES
00321             qDebug() << Q_FUNC_INFO << QString::fromUtf8("Retour %1 pour la requete : %2").arg(
QString::number(retour)).arg(requete);
00322             #endif
00323             if(retour)
00324             {
00325                 // pour chaque enregistrement
00326                 while ( r.next() )
00327                 {
00328                     // on récupère sous forme de QString la valeur du champs sélectionné
00329                     data = r.value(0).toString();
00330
00331                     #ifdef DEBUG_BASEDEDONNEES
00332                     //qDebug() << Q_FUNC_INFO << "Enregistrement -> " << data;
00333                     #endif
00334
00335                     // on stocke l'enregistrement dans le QVector

```

```

00336         donnees.push_back(data);
00337     }
00338     #ifdef DEBUG_BASEDEDONNEES
00339     qDebug() << Q_FUNC_INFO << "Enregistrement -> " << donnees;
00340     #endif
00341     return true;
00342 }
00343 else
00344 {
00345     qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : %1 pour la requête %2").arg(r.
lastError().text()).arg(requete);
00346     return false;
00347 }
00348 }
00349 else
00350 {
00351     qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : requête %1 non autorisée !").arg(requete
);
00352     return false;
00353 }
00354 }
00355 else
00356     return false;
00357 }

```

9.8.2.20 recuperer() [4/8]

```

bool BaseDeDonnees::recuperer (
    QString requete,
    QVector< QStringList > & donnees )

```

Références [db](#), et [mutex](#).

```

00363 {
00364     QMutexLocker verrou(&mutex);
00365     QSqlQuery r;
00366     bool retour;
00367     QStringList data;
00368
00369     if(db.isOpen())
00370     {
00371         if(requete.contains("SELECT"))
00372         {
00373             retour = r.exec(requete);
00374             #ifdef DEBUG_BASEDEDONNEES
00375             qDebug() << Q_FUNC_INFO << QString::fromUtf8("Retour %1 pour la requete : %2").arg(
QString::number(retour)).arg(requete);
00376             #endif
00377             if(retour)
00378             {
00379                 // pour chaque enregistrement
00380                 while ( r.next() )
00381                 {
00382                     // on récupère sous forme de QString la valeur de tous les champs sélectionnés
00383                     // et on les stocke dans une liste de QString
00384                     for(int i=0;i<r.record().count();i++)
00385                         data << r.value(i).toString();
00386
00387                     #ifdef DEBUG_BASEDEDONNEES
00388                     //qDebug() << Q_FUNC_INFO << "Enregistrement -> " << data;
00389                     /*for(int i=0;i<r.record().count();i++)
00390                         qDebug() << r.value(i).toString();*/
00391                     #endif
00392
00393                     // on stocke l'enregistrement dans le QVector
00394                     donnees.push_back(data);
00395
00396                     // on efface la liste de QString pour le prochain enregistrement
00397                     data.clear();
00398                 }
00399             #ifdef DEBUG_BASEDEDONNEES
00400             qDebug() << Q_FUNC_INFO << "Enregistrement -> " << donnees;
00401             #endif
00402             return true;
00403         }
00404     else
00405     {
00406         qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : %1 pour la requête %2").arg(r.
lastError().text()).arg(requete);
00407         return false;

```

```
00408         }
00409     }
00410     else
00411     {
00412         qDebug() << Q_FUNC_INFO << QString::fromUtf8("Erreur : requête %1 non autorisée !").arg(requete
00413     );
00414         return false;
00415     }
00416     else
00417         return false;
00418 }
```

9.8.2.21 recuperer() [5/8]

```
Q_INVOKABLE bool BaseDeDonnees::recuperer (
    QString requete,
    QString & donnees )
```

9.8.2.22 recuperer() [6/8]

```
Q_INVOKABLE bool BaseDeDonnees::recuperer (
    QString requete,
    QStringList & donnees )
```

9.8.2.23 recuperer() [7/8]

```
Q_INVOKABLE bool BaseDeDonnees::recuperer (
    QString requete,
    QVector< QString > & donnees )
```

9.8.2.24 recuperer() [8/8]

```
Q_INVOKABLE bool BaseDeDonnees::recuperer (
    QString requete,
    QVector< QStringList > & donnees )
```

9.8.3 Documentation des données membres

9.8.3.1 baseDeDonnees

static [BaseDeDonnees](#) * BaseDeDonnees::baseDeDonnees = NULL [static], [private]

Référencé par [destruireInstance\(\)](#), et [getInstance\(\)](#).

9.8.3.2 db

`QSqlDatabase BaseDeDonnees::db [private]`

Référencé par [BaseDeDonnees\(\)](#), [connecter\(\)](#), [deconnecter\(\)](#), [estConnecte\(\)](#), [estOuvert\(\)](#), [executer\(\)](#), [ouvrir\(\)](#), et [recuperer\(\)](#).

9.8.3.3 mutex

`QMutex BaseDeDonnees::mutex [private]`

Référencé par [connecter\(\)](#), [deconnecter\(\)](#), [estConnecte\(\)](#), [estOuvert\(\)](#), [executer\(\)](#), [getNomBDD\(\)](#), [ouvrir\(\)](#), et [recuperer\(\)](#).

9.8.3.4 nbAcces

`static int BaseDeDonnees::nbAcces = 0 [static], [private]`

Référencé par [destruireInstance\(\)](#), et [getInstance\(\)](#).

9.8.3.5 nomBDD

`QString BaseDeDonnees::nomBDD [private]`

Référencé par [getNomBDD\(\)](#).

9.8.3.6 typeBase

`QString BaseDeDonnees::typeBase = "MYSQL" [static], [private]`

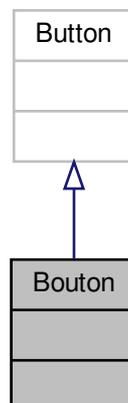
Référencé par [BaseDeDonnees\(\)](#), [connecter\(\)](#), et [ouvrir\(\)](#).

La documentation de cette classe a été générée à partir des fichiers suivants :

- [ihm-estock/baseDeDonnees.h](#)
- [ihm-estock/baseDeDonnees.cpp](#)

9.9 Référence de la classe Bouton

Graphe de collaboration de Bouton :



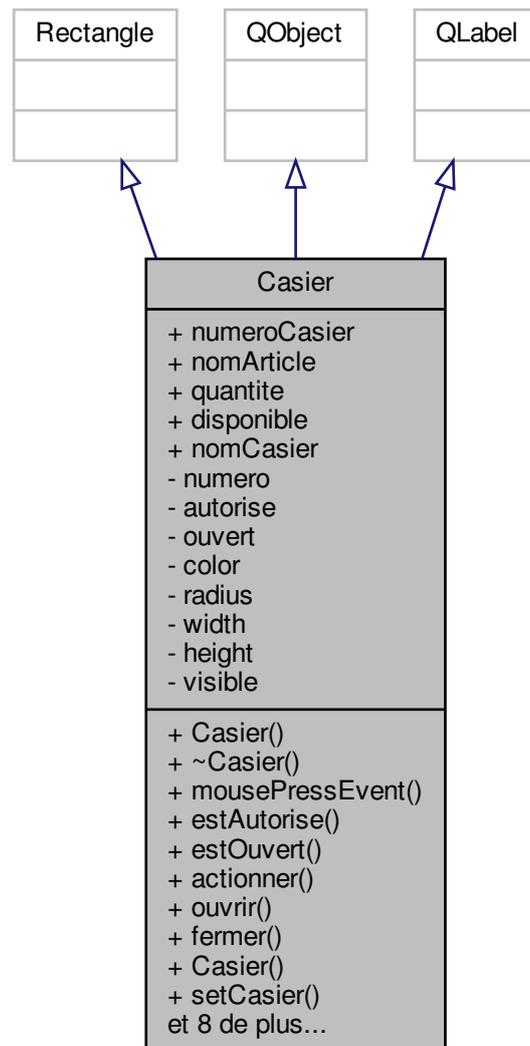
La documentation de cette classe a été générée à partir du fichier suivant :

- [Bouton.qml](#)

9.10 Référence de la classe Casier

```
#include <casier.h>
```

Grphe de collaboration de Casier :



Signaux

- void `clicked` (int)
- void `clicked` ()
- void `numeroCasierChanged` ()
- void `nomArticleChanged` ()
- void `quantiteChanged` ()
- void `disponibleChanged` ()

Fonctions membres publiques

- `Casier` (int `numero`, bool `autorise`=true, bool `present`=true, `QWidget` *`parent`=0)
- `~Casier` ()
- void `mousePressEvent` (`QMouseEvent` *`event`)
- bool `estAutorise` () const
- bool `estOuvert` () const
- void `actionner` ()
- void `ouvrir` ()
- void `fermer` ()
- `Casier` (int `numeroCasier`=0, `QString` `nomArticle`="", int `quantite`=0, int `disponible`=0, `QObject` *`parent`=nullptr)
- void `setCasier` (int `numeroCasier`, `QString` `nomArticle`, int `quantite`, int `disponible`)
- int `getNumeroCasier` () const
- void `setNumeroCasier` (int `numeroCasier`)
- `QString` `getNomArticle` () const
- void `setNomArticle` (`QString` `nomArticle`)
- int `getQuantite` () const
- void `setQuantite` (int `quantite`)
- int `getDisponible` () const
- void `setDisponible` (int `disponible`)

Propriétés

- int `numeroCasier`
- `QString` `nomArticle`
- int `quantite`
- int `disponible`
- string `nomCasier`

Attributs privés

- int `numero`
- bool `autorise`
- bool `ouvert`
- var `color`
- var `radius`
- var `width`
- var `height`
- var `visible`

9.10.1 Documentation des constructeurs et destructeur

9.10.1.1 `Casier()` [1/2]

```
Casier::Casier (
    int numero,
    bool autorise = true,
    bool present = true,
    QWidget * parent = 0 )
```

Références `numero`.

```
00004                                     : QLabel(parent),
    numero(numero), autorise(autorise), ouvert (
    ouvert)
00005 {
00006     qDebug() << Q_FUNC_INFO << numero;
00007     // choix par défaut
00008     setText("Casier " + QString::number(numero));
00009     if(autorise)
00010     {
00011         setStyleSheet("background-color: rgb(239, 41, 41);");
00012     }
00013     else
00014     {
00015         setStyleSheet("background-color: rgb(85, 85, 85);");
00016     }
00017
00018     setAlignment(Qt::AlignHCenter|Qt::AlignVCenter); // centré
00019     setContentsMargins(10, 0, 10, 0); // Marges : Gauche Haut Droite Bas
00020 }
```

9.10.1.2 ~Casier()

```
Casier::~Casier ( )
```

Références [numero](#).

```
00023 {
00024     qDebug() << Q_FUNC_INFO << numero;
00025 }
```

9.10.1.3 Casier() [2/2]

```
Casier::Casier (
    int numeroCasier = 0,
    QString nomArticle = "",
    int quantite = 0,
    int disponible = 0,
    QObject * parent = nullptr ) [explicit]
```

```
00004                                     : QObject
    (parent), numeroCasier(numeroCasier), nomArticle(
    nomArticle), quantite(quantite), disponible(
    disponible)
00005 {
00006
00007 }
```

9.10.2 Documentation des fonctions membres

9.10.2.1 actionner()

```
void Casier::actionner ( )
```

Références [autorise](#), [fermer\(\)](#), [ouvert](#), et [ouvrir\(\)](#).

```
00038 {
00039     if(!autorise)
00040         return;
00041     if(ouvert)
00042         fermer();
00043     else
00044         ouvrir();
00045 }
```

9.10.2.2 clicked [1/2]

```
void Casier::clicked (
    int ) [signal]
```

9.10.2.3 clicked [2/2]

```
void Casier::clicked ( ) [signal]
```

Référencé par [mousePressEvent\(\)](#).

9.10.2.4 disponibleChanged

```
void Casier::disponibleChanged ( ) [signal]
```

9.10.2.5 estAutorise()

```
bool Casier::estAutorise ( ) const
```

Références [autorise](#).

```
00028 {  
00029     return autorise;  
00030 }
```

9.10.2.6 estOuvert()

```
bool Casier::estOuvert ( ) const
```

Références [ouvert](#).

```
00033 {  
00034     return ouvert;  
00035 }
```

9.10.2.7 fermer()

```
void Casier::fermer ( )
```

A faire envoyer trame fermeture

Références [autorise](#), et [ouvert](#).

Référencé par [actionner\(\)](#).

```
00059 {  
00060     if(!autorise)  
00061         return;  
00065     setStyleSheet("background-color: rgb(239, 41, 41);");  
00066     ouvert = false;  
00067 }
```

9.10.2.8 getDisponible()

```
int Casier::getDisponible ( ) const
```

Références [disponible](#).

```
00038 {  
00039     return disponible;  
00040 }
```

9.10.2.9 getNomArticle()

```
QString Casier::getNomArticle ( ) const
```

Références [nomArticle](#).

```
00028 {  
00029     return nomArticle;  
00030 }
```

9.10.2.10 getNumeroCasier()

```
int Casier::getNumeroCasier ( ) const
```

Références [numeroCasier](#).

```
00018 {  
00019     return numeroCasier;  
00020 }
```

9.10.2.11 getQuantite()

```
int Casier::getQuantite ( ) const
```

Références [quantite](#).

```
00048 {  
00049     return quantite;  
00050 }
```

9.10.2.12 mousePressEvent()

```
void Casier::mousePressEvent (
    QMouseEvent * event )
```

Références [autorise](#), [clicked\(\)](#), et [numero](#).

```
00070 {
00071     if(!autorise)
00072         return;
00073
00074     if(event->button() == Qt::LeftButton)
00075     {
00076         if(autorise)
00077         {
00078             emit clicked(numero);
00079             emit clicked();
00080         }
00081     }
00082 }
```

9.10.2.13 nomArticleChanged

```
void Casier::nomArticleChanged ( ) [signal]
```

9.10.2.14 numeroCasierChanged

```
void Casier::numeroCasierChanged ( ) [signal]
```

9.10.2.15 ouvrir()

```
void Casier::ouvrir ( )
```

A faire envoyer trame ouverture

Références [autorise](#), et [ouvert](#).

Référencé par [actionner\(\)](#).

```
00048 {
00049     if(!autorise)
00050         return;
00054     setStyleSheet("background-color: rgb(115, 210, 22);");
00055     ouvert = true;
00056 }
```

9.10.2.16 quantiteChanged

```
void Casier::quantiteChanged ( ) [signal]
```

9.10.2.17 setCasier()

```
void Casier::setCasier (
    int numeroCasier,
    QString nomArticle,
    int quantite,
    int disponible )
```

Références [disponible](#), [nomArticle](#), [numeroCasier](#), et [quantite](#).

```
00010 {
00011     numeroCasier = numeroCasier;
00012     nomArticle = nomArticle;
00013     quantite = quantite;
00014     disponible = disponible;
00015 }
```

9.10.2.18 setDisponible()

```
void Casier::setDisponible (
    int disponible )
```

Références [disponible](#).

```
00043 {
00044     disponible = disponible;
00045 }
```

9.10.2.19 setNomArticle()

```
void Casier::setNomArticle (
    QString nomArticle )
```

Références [nomArticle](#).

```
00033 {
00034     nomArticle = nomArticle;
00035 }
```

9.10.2.20 setNumeroCasier()

```
void Casier::setNumeroCasier (
    int numeroCasier )
```

Références [numeroCasier](#).

```
00023 {
00024     numeroCasier = numeroCasier;
00025 }
```

9.10.2.21 setQuantite()

```
void Casier::setQuantite (  
    int quantite )
```

Références [quantite](#).

```
00053 {  
00054     quantite = quantite;  
00055 }
```

9.10.3 Documentation des données membres

9.10.3.1 autorise

```
bool Casier::autorise [private]
```

Référencé par [actionner\(\)](#), [estAutorise\(\)](#), [fermer\(\)](#), [mousePressEvent\(\)](#), et [ouvrir\(\)](#).

9.10.3.2 color

```
var Casier::color [private]
```

9.10.3.3 height

```
var Casier::height [private]
```

9.10.3.4 numero

```
int Casier::numero [private]
```

Référencé par [Casier\(\)](#), [mousePressEvent\(\)](#), et [~Casier\(\)](#).

9.10.3.5 ouvert

```
bool Casier::ouvert [private]
```

Référencé par [actionner\(\)](#), [estOuvert\(\)](#), [fermer\(\)](#), et [ouvrir\(\)](#).

9.10.3.6 radius

```
var Casier::radius [private]
```

9.10.3.7 visible

```
var Casier::visible [private]
```

9.10.3.8 width

```
var Casier::width [private]
```

9.10.4 Documentation des propriétés

9.10.4.1 disponible

```
int Casier::disponible [read], [write]
```

Référencé par [getDisponible\(\)](#), [setCasier\(\)](#), et [setDisponible\(\)](#).

9.10.4.2 nomArticle

```
QString Casier::nomArticle [read], [write]
```

Référencé par [getNomArticle\(\)](#), [setCasier\(\)](#), et [setNomArticle\(\)](#).

9.10.4.3 nomCasier

```
string Casier::nomCasier
```

9.10.4.4 numeroCasier

```
int Casier::numeroCasier [read], [write]
```

Référencé par [getNumeroCasier\(\)](#), [setCasier\(\)](#), et [setNumeroCasier\(\)](#).

9.10.4.5 quantite

```
int Casier::quantite [read], [write]
```

Référencé par [getQuantite\(\)](#), [setCasier\(\)](#), et [setQuantite\(\)](#).

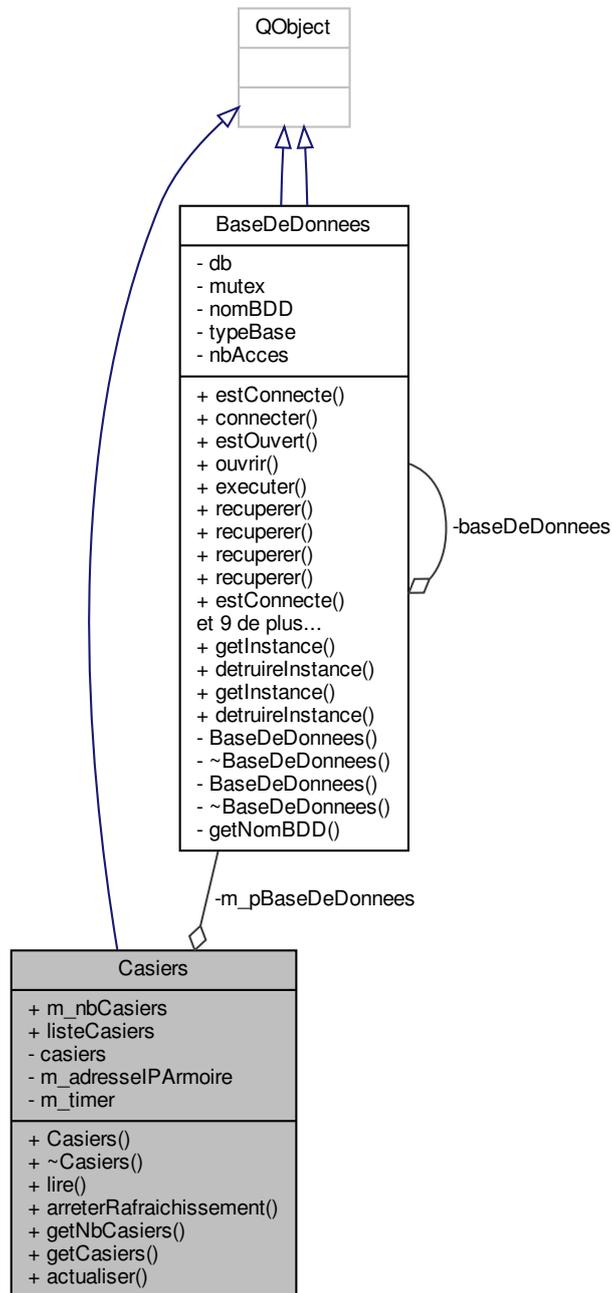
La documentation de cette classe a été générée à partir des fichiers suivants :

- [ihm-estock/casier.h](#)
- [Casier.qml](#)
- [ihm-estock/casier.cpp](#)

9.11 Référence de la classe Casiers

```
#include <casiers.h>
```

Graphe de collaboration de Casiers :



Connecteurs publics

— void [actualiser](#) ()

Signaux

- void [listeCasiersChanged](#) ()
- void [nbCasiersChanged](#) ()

Fonctions membres publiques

- [Casiers](#) (QObject *parent=nullptr)
- [~Casiers](#) ()
- Q_INVOKABLE QString [lire](#) (QString adresseIPArmoire, bool debug=true)
- Q_INVOKABLE void [arreterRafraichissement](#) ()
- QString [getNbCasiers](#) ()
- QVariant [getCasiers](#) ()

Propriétés

- QString [m_nbCasiers](#)
- QVariant [listeCasiers](#)

Attributs privés

- QList< QObject * > [casiers](#)
- [BaseDeDonnees](#) * [m_pBaseDeDonnees](#)
- QString [m_adresseIPArmoire](#)
- QTimer * [m_timer](#)

9.11.1 Documentation des constructeurs et destructeur

9.11.1.1 Casiers()

```
Casiers::Casiers (
    QObject * parent = nullptr ) [explicit]
```

Références [actualiser\(\)](#), [BaseDeDonnees : :getInstance\(\)](#), [m_pBaseDeDonnees](#), et [m_timer](#).

```
00005         : QObject (parent)
00006 {
00007     m_pBaseDeDonnees = BaseDeDonnees::getInstance ("MYSQL");
00008     m_timer = new QTimer (this);
00009     connect (m_timer, SIGNAL (timeout ()), this, SLOT (actualiser ()));
00010 }
```

9.11.1.2 ~Casiers()

```
Casiers::~Casiers ( )
```

Références [BaseDeDonnees : :destruireInstance\(\)](#), et [m_timer](#).

```
00013 {
00014     if (m_timer->isActive ())
00015         m_timer->stop ();
00016     BaseDeDonnees::destruireInstance ("MYSQL");
00017 }
```

9.11.2 Documentation des fonctions membres

9.11.2.1 actualiser

```
void Casiers::actualiser ( ) [slot]
```

Références [lire\(\)](#), et [m_adresseIPArmoire](#).

Référencé par [Casiers\(\)](#).

```
00070 {  
00071     if (!m_adresseIPArmoire.isEmpty())  
00072         lire(m_adresseIPArmoire, false);  
00073 }
```

9.11.2.2 arreterRafrachissement()

```
void Casiers::arreterRafrachissement ( )
```

Références [m_timer](#).

```
00020 {  
00021     if (m_timer->isActive())  
00022         m_timer->stop();  
00023 }
```

9.11.2.3 getCasiers()

```
QVariant Casiers::getCasiers ( )
```

Références [casiers](#).

```
00031 {  
00032     return QVariant::fromValue(casiers);  
00033 }
```

9.11.2.4 getNbCasiers()

```
QString Casiers::getNbCasiers ( )
```

Références [m_nbCasiers](#).

```
00026 {  
00027     return m_nbCasiers;  
00028 }
```

9.11.2.5 lire()

```
QString Casiers::lire (
    QString adresseIPArmoire,
    bool debug = true )
```

Références [casiers](#), [listeCasiersChanged\(\)](#), [m_adresseIPArmoire](#), [m_nbCasiers](#), [m_pBaseDeDonnees](#), [m_timer](#), et [BaseDeDonnees : :recuperer\(\)](#).

Référencé par [actualiser\(\)](#).

```
00036 {
00037     this->m_adresseIPArmoire = adresseIPArmoire;
00038     QString requeteNbCasiers = "SELECT Armoire.nbCasiers FROM Armoire WHERE Armoire.AdresseIP = '" +
    adresseIPArmoire + "'";
00039     m_pBaseDeDonnees->recuperer(requeteNbCasiers,
    m_nbCasiers);
00040
00041     QString requeteStockCasier = "SELECT Stock.NumeroCasier, Article.Nom, Stock.Quantite, Stock.Disponible
    FROM Stock INNER JOIN Article ON Stock.idArticle = Article.idArticle ORDER BY Stock.NumeroCasier ASC";
00042     QVector<QStringList> enregistrementsCasiers;
00043
00044     casiers.clear();
00045     if (m_pBaseDeDonnees->recuperer(requeteStockCasier,enregistrementsCasiers))
00046     {
00047         QStringList champsCasier;
00048
00049         for(int i = 0; i < enregistrementsCasiers.size(); i++)
00050         {
00051             champsCasier = enregistrementsCasiers.at(i);
00052             if(debug)
00053                 qDebug() << Q_FUNC_INFO << "champsCasier" << champsCasier;
00054
00055             Casier *pCasier = new Casier(champsCasier.at(0).toInt(), champsCasier.at(1),
    champsCasier.at(2).toInt(), champsCasier.at(3).toInt());
00056             casiers.append(pCasier);
00057         }
00058         emit listeCasiersChanged();
00059
00060         return m_nbCasiers;
00061     }
00062
00063     if(!m_timer->isActive())
00064         m_timer->start(500);
00065
00066     return m_nbCasiers;
00067 }
```

9.11.2.6 listeCasiersChanged

```
void Casiers::listeCasiersChanged ( ) [signal]
```

Référencé par [lire\(\)](#).

9.11.2.7 nbCasiersChanged

```
void Casiers::nbCasiersChanged ( ) [signal]
```

9.11.3 Documentation des données membres

9.11.3.1 casiers

```
QList<QObject*> Casiers::casiers [private]
```

Référencé par [getCasiers\(\)](#), et [lire\(\)](#).

9.11.3.2 m_adresseIPArmoire

```
QString Casiers::m_adresseIPArmoire [private]
```

Référencé par [actualiser\(\)](#), et [lire\(\)](#).

9.11.3.3 m_pBaseDeDonnees

```
BaseDeDonnees* Casiers::m_pBaseDeDonnees [private]
```

Référencé par [Casiers\(\)](#), et [lire\(\)](#).

9.11.3.4 m_timer

```
QTimer* Casiers::m_timer [private]
```

Référencé par [arreterRafraichissement\(\)](#), [Casiers\(\)](#), [lire\(\)](#), et [~Casiers\(\)](#).

9.11.4 Documentation des propriétés

9.11.4.1 listeCasiers

```
QVariant Casiers::listeCasiers [read]
```

9.11.4.2 m_nbCasiers

```
QString Casiers::m_nbCasiers [read]
```

Référencé par [getNbCasiers\(\)](#), et [lire\(\)](#).

La documentation de cette classe a été générée à partir des fichiers suivants :

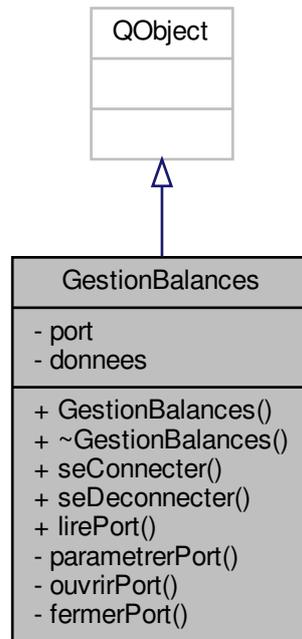
- [casiers.h](#)
- [casiers.cpp](#)

9.12 Référence de la classe GestionBalances

Déclaration de la classe [GestionBalances](#).

```
#include <GestionBalances.h>
```

Grphe de collaboration de GestionBalances :



Connecteurs publics

- void [lirePort](#) ()
Réception des trames balance.

Fonctions membres publiques

- [GestionBalances](#) (QObject *parent=nullptr)
Constructeur.
- [~GestionBalances](#) ()
Destructeur de la calsse [GestionBalances](#).
- void [seConnecter](#) ()
Ouvre et paramètre le port.
- void [seDeconnecter](#) ()
Fermeture du port.

Fonctions membres privées

- void [paramettrerPort](#) ()
Paramétrage du port.
- void [ouvrirPort](#) ()
Ouverture du port.
- void [fermerPort](#) ()
Fermeture du port.

Attributs privés

- `QSerialPort` * [port](#)
- `QByteArray` [donnees](#)

9.12.1 Description détaillée**Auteur**

Hadrien Gimenez

Version

1.1

9.12.2 Documentation des constructeurs et destructeur**9.12.2.1 GestionBalances()**

```
GestionBalances::GestionBalances (
    QObject * parent = nullptr ) [explicit]
```

Paramètres

<i>parent</i>	QObject Adresse de l'objet Qt parent
---------------	--------------------------------------

Références [PORT](#), [port](#), et [seConnecter\(\)](#).

```
00012                                     : QObject (parent)
00013 {
00014     port = new QSerialPort (PORT, this);
00015
00016     this->seConnecter ();
00017 }
```

9.12.2.2 ~GestionBalances()

```
GestionBalances::~GestionBalances ( )
```

Références [seDeconnecter\(\)](#).

```
00026 {
00027     this->seDeconnecter ();
00028 }
```

9.12.3 Documentation des fonctions membres

9.12.3.1 fermerPort()

```
void GestionBalances::fermerPort ( ) [private]
```

Références [port](#).

Référencé par [seDeconnecter\(\)](#).

```
00087 {
00088     if(port->isOpen())
00089         port->close();
00090     qDebug() << Q_FUNC_INFO << "etat ouverture port" << port->isOpen();
00091 }
```

9.12.3.2 lirePort

```
void GestionBalances::lirePort ( ) [slot]
```

Références [donnees](#), et [port](#).

Référencé par [ouvrirPort\(\)](#).

```
00067 {
00068     if(port->canReadLine())
00069     {
00070         QByteArray donnees = port->readLine();
00071         qDebug() << Q_FUNC_INFO << donnees;
00072         if(donnees.startsWith("$"))
00073         {
00074             qDebug() << Q_FUNC_INFO << donnees;
00075             donnees.clear();
00076         }
00077     }
00078 }
```

9.12.3.3 ouvrirPort()

```
void GestionBalances::ouvrirPort ( ) [private]
```

Références [lirePort\(\)](#), et [port](#).

Référencé par [seConnecter\(\)](#).

```
00051 {
00052     port->open(QIODevice::ReadOnly);
00053     qDebug() << Q_FUNC_INFO << "etat ouverture port" << port->isOpen();
00054     if(port->isOpen())
00055     {
00056         connect(port, SIGNAL(readyRead()), this, SLOT(lirePort()));
00057     }
00058 }
```

9.12.3.4 paramererPort()

```
void GestionBalances::paramererPort ( ) [private]
```

Références [port](#).

Référencé par [seConnecter\(\)](#).

```
00037 {  
00038     port->setBaudRate(QSerialPort::Baud9600);  
00039     port->setDataBits(QSerialPort::Data8);  
00040     port->setParity(QSerialPort::NoParity);  
00041     port->setStopBits(QSerialPort::OneStop);  
00042     port->setFlowControl(QSerialPort::NoFlowControl);  
00043 }
```

9.12.3.5 seConnecter()

```
void GestionBalances::seConnecter ( )
```

Références [ouvrirPort\(\)](#), et [paramererPort\(\)](#).

Référencé par [GestionBalances\(\)](#).

```
00100 {  
00101     this->ouvrirPort();  
00102     this->paramererPort();  
00103 }
```

9.12.3.6 seDeconnecter()

```
void GestionBalances::seDeconnecter ( )
```

Références [fermerPort\(\)](#).

Référencé par [~GestionBalances\(\)](#).

```
00112 {  
00113     this->fermerPort();  
00114 }
```

9.12.4 Documentation des données membres

9.12.4.1 donnees

```
QByteArray GestionBalances::donnees [private]
```

Référencé par [lirePort\(\)](#).

9.12.4.2 port

```
QSerialPort* GestionBalances::port [private]
```

Référencé par [fermerPort\(\)](#), [GestionBalances\(\)](#), [lirePort\(\)](#), [ouvrirPort\(\)](#), et [parametrerPort\(\)](#).

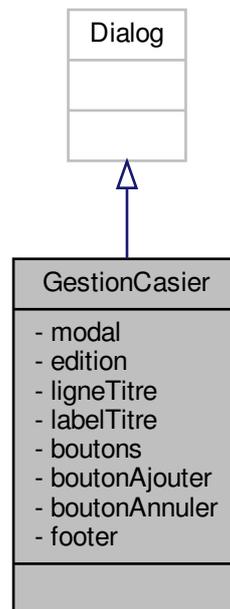
La documentation de cette classe a été générée à partir des fichiers suivants :

- [GestionBalances.h](#)
- [GestionBalances.cpp](#)

9.13 Référence de la classe GestionCasier

La boîte de dialogue [GestionCasier](#).

Graphe de collaboration de GestionCasier :



Attributs privés

- var [modal](#)
- Column [edition](#)
Conteneur des éléments de la page qml [GestionCasier](#).
- Row [ligneTitre](#)
Conteneur du texte [labelTitre](#).
- Text [labelTitre](#)
titre de la boîte de dialogue
- Row [boutons](#)
Conteneur des boutons [boutonAjouter](#) et [boutonAnnuler](#).
- Button [boutonAjouter](#)
[Bouton](#) pour ajouter une armoire.
- Button [boutonAnnuler](#)
[Bouton](#) pour revenir à la page [IHMPricipale](#).
- var [footer](#)
 pied de page

9.13.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

9.13.2 Documentation des données membres

9.13.2.1 boutonAjouter

Button GestionCasier::boutonAjouter [private]

9.13.2.2 boutonAnnuler

Button GestionCasier::boutonAnnuler [private]

9.13.2.3 boutons

Row GestionCasier::boutons [private]

9.13.2.4 edition

Column GestionCasier::edition [private]

9.13.2.5 footer

var GestionCasier::footer [private]

9.13.2.6 labelTitre

Text GestionCasier::labelTitre [private]

9.13.2.7 ligneTitre

Row GestionCasier::ligneTitre [private]

9.13.2.8 modal

```
var GestionCasier::modal [private]
```

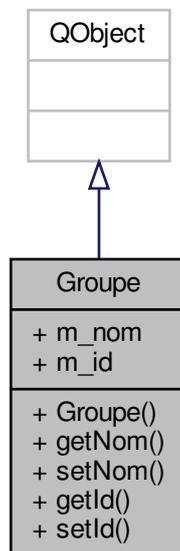
La documentation de cette classe a été générée à partir du fichier suivant :

— [GestionCasier.qml](#)

9.14 Référence de la classe Groupe

```
#include <groupe.h>
```

Grappe de collaboration de Groupe :



Signaux

— void [nomChanged](#) ()
 — void [idChanged](#) ()

Fonctions membres publiques

— [Groupe](#) (QString p_id="", QString p_nom="", QObject *parent=nullptr)
 — QString [getNom](#) () const
 — void [setNom](#) (QString p_nom)
 — QString [getId](#) () const
 — void [setId](#) (QString p_id)

Propriétés

— QString [m_nom](#)
 — QString [m_id](#)

9.14.1 Documentation des constructeurs et destructeur

9.14.1.1 Groupe()

```
Groupe::Groupe (
    QString p_id = "",
    QString p_nom = "",
    QObject * parent = nullptr ) [explicit]

00003                                     : QObject (parent),
    m_nom(p_nom), m_id(p_id)
00004 {
00005 }
```

9.14.2 Documentation des fonctions membres

9.14.2.1 getId()

```
QString Groupe::getId ( ) const
```

Références [m_id](#).

```
00018 {
00019     return m_id;
00020 }
```

9.14.2.2 getNom()

```
QString Groupe::getNom ( ) const
```

Références [m_nom](#).

```
00008 {
00009     return m_nom;
00010 }
```

9.14.2.3 idChanged

```
void Groupe::idChanged ( ) [signal]
```

9.14.2.4 nomChanged

```
void Groupe::nomChanged ( ) [signal]
```

9.14.2.5 setId()

```
void Groupe::setId (
    QString p_id )
```

Références [m_id](#).

```
00023 {
00024     m_id = p_id;
00025 }
```

9.14.2.6 setNom()

```
void Groupe::setNom (
    QString p_nom )
```

Références [m_nom](#).

```
00013 {
00014     m_nom = p_nom;
00015 }
```

9.14.3 Documentation des propriétés

9.14.3.1 m_id

```
QString Groupe::m_id [read], [write]
```

Référencé par [getId\(\)](#), et [setId\(\)](#).

9.14.3.2 m_nom

```
QString Groupe::m_nom [read], [write]
```

Référencé par [getNom\(\)](#), et [setNom\(\)](#).

La documentation de cette classe a été générée à partir des fichiers suivants :

- [groupe.h](#)
- [groupe.cpp](#)

- *Permet de valide l'accès à l'utilisateur pour le connecter à l'armoire.*
- void [authentifierSansBadge](#) ()
- void [autoriserSansBadge](#) (QString identifiant, QString motDePasse)
- *Permet à un utilisateur de s'authentifier sans le badge.*
- void [effacerMessage](#) ()
- void [passerSansBadge](#) ()

Fonctions membres publiques

- [IhmAuthentification](#) (QWidget *parent=nullptr)
- *Constructeur de la fenêtre principale.*
- [~IhmAuthentification](#) ()
- *Destructeur de la fenêtre principale.*

Fonctions membres privées

- void [initialiserFenetre](#) ()
- *Initialise le mode plein écran.*
- void [afficherAuthentificationNonValide](#) ()
- *Affiche le message d'erreur du badge non valide.*
- void [reinitialiserSaisieBadge](#) ()
- *Permet de reinitialiser la saisie du badge.*

Attributs privés

- Ui : :IhmAuthentification * ui
- *agrégation de la définition de la fenêtre UI*
- [AuthentificationBadge](#) * [authentificationBadge](#)
- [IhmAuthentificationSansBadge](#) * [ihmAuthentificationSansBadge](#)
- [IhmGestionArmoire](#) * [ihmGestionArmoire](#)
- [Utilisateur](#) * [utilisateur](#)
- [Utilisateur](#) * [administrateur](#)
- [BaseDeDonnees](#) * [bdd](#)
- *agrégation de l'objet [BaseDeDonnees](#)*

9.15.1 Description détaillée

Auteur

Thomas MACHON

Version

1.1

9.15.2 Documentation des constructeurs et destructeur

9.15.2.1 IhmAuthentification()

```
IhmAuthentification::IhmAuthentification (
    QWidget * parent = nullptr ) [explicit]
```

Paramètres

<i>parent</i>	QObject Adresse de l'objet Qt parent
---------------	--------------------------------------

Références [authentificationBadge](#), [authentifierSansBadge\(\)](#), [bdd](#), [BaseDeDonnees : :connecter\(\)](#), [BaseDeDonnees : :estConnecte\(\)](#), [BaseDeDonnees : :getInstance\(\)](#), [ihmAuthentificationSansBadge](#), [ihmGestionArmoire](#), [initialiserFenetre\(\)](#), [passerSansBadge\(\)](#), [ui](#), et [validerAcces\(\)](#).

```

00031                                     :
00032     QWidget(parent),
00033     ui(new Ui::IhmAuthentification), utilisateur(nullptr)
00034 {
00035     ui->setupUi(this);
00036     qDebug() << Q_FUNC_INFO;
00037
00038     bdd = BaseDeDonnees::getInstance();
00039     if(!bdd->estConnecte())
00040         bdd->connecter();
00041
00042     ihmAuthentificationSansBadge = nullptr;
00043     authentificationBadge = new AuthentificationBadge(this);
00044
00045     ihmGestionArmoire = new IhmGestionArmoire(this);
00046     ihmGestionArmoire->hide();
00047
00048     // Connexion des signaux/slots
00049     connect(ui->saisieBadge, SIGNAL(editingFinished()), this, SLOT(
00050     validerAcces()));
00051     connect(ui->boutonPasDeBadge, SIGNAL(clicked()), this, SLOT(
00052     authentifierSansBadge()));
00053     // mode développeur seulement
00054     connect(ui->boutonPasser, SIGNAL(clicked()), this, SLOT(passerSansBadge()));
00055 }

```

9.15.2.2 ~IhmAuthentification()

```
IhmAuthentification::~IhmAuthentification ( )
```

Références [BaseDeDonnees : :destruireInstance\(\)](#), et [ui](#).

```

00064 {
00065     BaseDeDonnees::destruireInstance();
00066     delete ui;
00067     qDebug() << Q_FUNC_INFO;
00068 }

```

9.15.3 Documentation des fonctions membres

9.15.3.1 afficherAuthentificationNonValide()

```
void IhmAuthentification::afficherAuthentificationNonValide ( ) [private]
```

Références [effacerMessage\(\)](#), et [ui](#).

Référencé par [validerAcces\(\)](#).

```

00105 {
00106     ui->labelBadgeNonValide->setText("Badge non valide !");
00107     QTimer::singleShot(1000, this, SLOT(effacerMessage()));
00108 }

```

9.15.3.2 authentifierSansBadge

```
void IhmAuthentification::authentifierSansBadge ( ) [slot]
```

Références [autoriserSansBadge\(\)](#), et [ihmAuthentificationSansBadge](#).

Référencé par [IhmAuthentification\(\)](#).

```
00154 {
00155     if(ihmAuthentificationSansBadge == nullptr)
00156     {
00157         ihmAuthentificationSansBadge = new
IhmAuthentificationSansBadge(this);
00158         connect(ihmAuthentificationSansBadge, SIGNAL(utilisateurAutorise(
QString,QString)), this, SLOT(autoriserSansBadge(QString,QString)));
00159     }
00160     ihmAuthentificationSansBadge->show();
00161 }
```

9.15.3.3 autoriserSansBadge

```
void IhmAuthentification::autoriserSansBadge (
    QString identifiant,
    QString motDePasse ) [slot]
```

Références [IhmAuthentificationSansBadge : :getIdUtilisateur\(\)](#), [ihmAuthentificationSansBadge](#), [ihmGestionArmoire](#), [IhmGestionArmoire : :setUtilisateur\(\)](#), et [utilisateur](#).

Référencé par [authentifierSansBadge\(\)](#).

```
00170 {
00171     qDebug() << Q_FUNC_INFO << "identifiant" << identifiant << "motDePasse" << motDePasse;
00172     utilisateur = new Utilisateur(
ihmAuthentificationSansBadge->getIdUtilisateur());
00173     ihmGestionArmoire->setUtilisateur(utilisateur);
00174     ihmGestionArmoire->show();
00175 }
```

9.15.3.4 effacerMessage

```
void IhmAuthentification::effacerMessage ( ) [slot]
```

Références [ui](#).

Référencé par [afficherAuthentificationNonValide\(\)](#).

```
00186 {
00187     ui->labelBadgeNonValide->setText("");
00188 }
```

9.15.3.5 initialiserFenetre()

```
void IhmAuthentification::initialiserFenetre ( ) [private]
```

Références [IHM_CSS](#), et [ui](#).

Référencé par [IhmAuthentification\(\)](#).

```
00077 {
00078     setStyleSheet(IHM_CSS);
00079
00080     #ifndef DESKTOP
00081     const int w = qApp->desktop()->availableGeometry(this).width();
00082     const int h = qApp->desktop()->availableGeometry(this).height();
00083     resize(w, h);
00084     #else
00085     // 800x480
00086     resize(800, 480);
00087     #endif
00088     setWindowFlags(Qt::FramelessWindowHint | Qt::Dialog);
00089     //setWindowFlags(Qt::FramelessWindowHint | Qt::WindowStaysOnTopHint);
00090     //showMaximized();
00091     setWindowModality(Qt::WindowModal);
00092
00093     // Configuration de la saisie badge
00094     ui->saisieBadge->setFixedSize(0, 0);
00095     ui->saisieBadge->setFocus();
00096 }
```

9.15.3.6 passerSansBadge

```
void IhmAuthentification::passerSansBadge ( ) [slot]
```

Références [ihmGestionArmoire](#), [IhmGestionArmoire : :setUtilisateur\(\)](#), et [utilisateur](#).

Référencé par [IhmAuthentification\(\)](#).

```
00178 {
00179     utilisateur = new Utilisateur("1");
00180     ihmGestionArmoire->setUtilisateur(utilisateur);
00181     ihmGestionArmoire->show();
00182 }
```

9.15.3.7 reinitialiserSaisieBadge()

```
void IhmAuthentification::reinitialiserSaisieBadge ( ) [private]
```

Références [ui](#).

Référencé par [validerAcces\(\)](#).

```
00117 {
00118     // Réinitialiser la saisie d'un nouveau badge
00119     ui->saisieBadge->setText("");
00120     ui->saisieBadge->setFocus();
00121 }
```

9.15.3.8 validerAcces

```
IhmAuthentification::validerAcces ( ) [slot]
```

Références [afficherAuthentificationNonValide\(\)](#), [authentificationBadge](#), [AuthentificationBadge : :estAutorise\(\)](#), [AuthentificationBadge : :getIdUtilisateur\(\)](#), [ihmGestionArmoire](#), [AuthentificationBadge : :lireBadge\(\)](#), [reinitialiserSaisieBadge\(\)](#), [IhmGestionArmoire : :setUtilisateur\(\)](#), [ui](#), et [utilisateur](#).

Référencé par [IhmAuthentification\(\)](#).

```
00130 {
00131     if(authentificationBadge->lireBadge(ui->saisieBadge->text()))
00132     {
00133         if(authentificationBadge->estAutorise())
00134         {
00135             utilisateur = new Utilisateur(
00136             authentificationBadge->getIdUtilisateur());
00137             ihmGestionArmoire->setUtilisateur(
00138             utilisateur);
00139             ihmGestionArmoire->show();
00140         }
00141     }
00142     else
00143     {
00144         afficherAuthentificationNonValide();
00145     }
00146     reinitialiserSaisieBadge();
00147 }
```

9.15.4 Documentation des données membres

9.15.4.1 administrateur

```
Utilisateur* IhmAuthentification::administrateur [private]
```

9.15.4.2 authentificationBadge

```
AuthentificationBadge* IhmAuthentification::authentificationBadge [private]
```

Référencé par [IhmAuthentification\(\)](#), et [validerAcces\(\)](#).

9.15.4.3 bdd

```
BaseDeDonnees* IhmAuthentification::bdd [private]
```

Référencé par [IhmAuthentification\(\)](#).

9.15.4.4 ihmAuthentificationSansBadge

```
IhmAuthentificationSansBadge* IhmAuthentification::ihmAuthentificationSansBadge [private]
```

Référencé par [authentifierSansBadge\(\)](#), [autoriserSansBadge\(\)](#), et [IhmAuthentification\(\)](#).

9.15.4.5 ihmGestionArmoire

```
IhmGestionArmoire* IhmAuthentification::ihmGestionArmoire [private]
```

Référencé par [autoriserSansBadge\(\)](#), [IhmAuthentification\(\)](#), [passerSansBadge\(\)](#), et [validerAcces\(\)](#).

9.15.4.6 ui

```
Ui::IhmAuthentification* IhmAuthentification::ui [private]
```

Référencé par [afficherAuthentificationNonValide\(\)](#), [effacerMessage\(\)](#), [IhmAuthentification\(\)](#), [initialiserFenetre\(\)](#), [reinitialiserSaisieBadge\(\)](#), [validerAcces\(\)](#), et [~IhmAuthentification\(\)](#).

9.15.4.7 utilisateur

```
Utilisateur* IhmAuthentification::utilisateur [private]
```

Référencé par [autoriserSansBadge\(\)](#), [passerSansBadge\(\)](#), et [validerAcces\(\)](#).

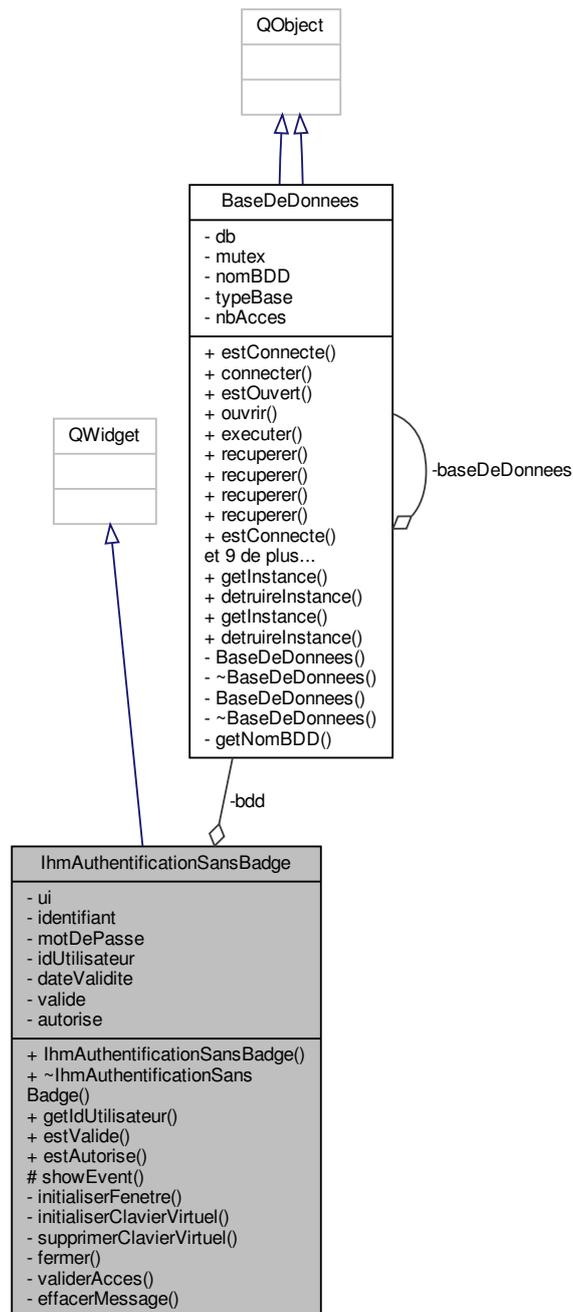
La documentation de cette classe a été générée à partir des fichiers suivants :

- [IhmAuthentification.h](#)
- [IhmAuthentification.cpp](#)

9.16 Référence de la classe IhmAuthentificationSansBadge

```
#include <IhmAuthentificationSansBadge.h>
```

Graphe de collaboration de IhmAuthenticationSansBadge :



Signaux

- void `utilisateurAutorise` (QString `identifiant`, QString `motDePasse`)

Fonctions membres publiques

- `IhmAuthenticationSansBadge` (QWidget *parent=nullptr)
Constructeur de la fenêtre d'authentification sans badge.
- `~IhmAuthenticationSansBadge` ()

- *Destructeur de la fenêtre d'authentification sans badge.*
- QString `getIdUtilisateur ()` const
Retourne l'identifiant de l'utilisateur.
- bool `estValide ()`
Requete SQL pour valider l'identifiant de l'utilisateur et son mot de passe.
- bool `estAutorise ()`

Fonctions membres protégées

- void `showEvent (QShowEvent *ev)`

Connecteurs privés

- void `fermer ()`
Permet de fermer le clavier virtuel.
- void `validerAcces ()`
Permet de valider l'accès à l'utilisateur avec son identifiant, mot de passe.
- void `effacerMessage ()`

Fonctions membres privées

- void `initialiserFenetre ()`
Initialise le mode plein écran.
- void `initialiserClavierVirtuel ()`
Initialisation d'un clavier virtuel.
- void `supprimerClavierVirtuel ()`
Suppression du clavier virtuel.

Attributs privés

- Ui : :IhmAuthentificationSansBadge * `ui`
- QString `identifiant`
le login de l'utilisateur
- QString `motDePasse`
- QString `idUtilisateur`
- QString `dateValidite`
- bool `valide`
- bool `autorise`
- BaseDeDonnees * `bdd`
agrégation de l'objet BaseDeDonnees

9.16.1 Documentation des constructeurs et destructeur

9.16.1.1 IhmAuthentificationSansBadge()

```
IhmAuthentificationSansBadge::IhmAuthentificationSansBadge (
    QWidget * parent = nullptr ) [explicit]
```

Constructeur de la fenêtre gestion armoire.

Paramètres

<i>parent</i>	QObject Adresse de l'objet Qt parent
---------------	--------------------------------------

Références [bdd](#), [BaseDeDonnees : :connecter\(\)](#), [BaseDeDonnees : :estConnecte\(\)](#), [fermer\(\)](#), [BaseDeDonnees : :getInstance\(\)](#), [initialiserFenetre\(\)](#), [ui](#), et [validerAcces\(\)](#).

```

00027                                     :
00028     QWidget(parent),
00029     ui(new Ui::IhmAuthentificationSansBadge)
00030 {
00031     ui->setupUi(this);
00032     qDebug() << Q_FUNC_INFO;
00033
00034     bdd = BaseDeDonnees::getInstance();
00035     if(!bdd->estConnecte())
00036         bdd->connecter();
00037
00038     // Connexion des signaux/slots
00039     connect(ui->boutonConnexion, SIGNAL(clicked()), this, SLOT(validerAcces()));
00040     connect(ui->boutonFermeture, SIGNAL(clicked()), this, SLOT(fermer()));
00041
00042     initialiserFenetre();
00043 }

```

9.16.1.2 ~IhmAuthentificationSansBadge()

```
IhmAuthentificationSansBadge::~IhmAuthentificationSansBadge ( )
```

Références [BaseDeDonnees : :destruireInstance\(\)](#), et [ui](#).

```

00051 {
00052     delete ui;
00053     BaseDeDonnees::destruireInstance();
00054     qDebug() << Q_FUNC_INFO;
00055 }

```

9.16.2 Documentation des fonctions membres

9.16.2.1 effacerMessage

```
void IhmAuthentificationSansBadge::effacerMessage ( ) [private], [slot]
```

Références [ui](#).

Référencé par [validerAcces\(\)](#).

```

00210 {
00211     ui->labelMessagAuthenticatioSansBadge->setText("");
00212 }

```

9.16.2.2 estAutorise()

```
bool IhmAuthentificationSansBadge::estAutorise ( )
```

Références [bdd](#), [dateValidite](#), [estValide\(\)](#), [identifiant](#), [idUtilisateur](#), [motDePasse](#), [BaseDeDonnees : :recuperer\(\)](#), [ui](#), et [valide](#).

Référencé par [validerAcces\(\)](#).

```
00188 {
00189     QString dateMaintenant = QDateTime::currentDateTime().toString("yyyy-MM-dd");
00190     identifiant = ui->lineEditAuthentification->text();
00191     motDePasse = ui->lineEditMotDePasse->text();
00192
00193     QString requete = "SELECT idUtilisateur, DateValidite FROM Utilisateur WHERE Identifiant='" +
ui->lineEditAuthentification->text() + "' AND MotDePasse='" + ui->lineEditMotDePasse->text() + "'";
00194     QStringList donneesUtilisateur;
00195     idUtilisateur.clear();
00196     dateValidite.clear();
00197     motDePasse.clear();
00198     valide = bdd->recuperer(requete, donneesUtilisateur);
00199     if (valide)
00200     {
00201         idUtilisateur = donneesUtilisateur.at(0);
00202         dateValidite = donneesUtilisateur.at(1);
00203         qDebug() << Q_FUNC_INFO << "valide" << valide << "idUtilisateur" <<
idUtilisateur << "dateValidite" << dateValidite << "motDePasse" <<
motDePasse;
00204     }
00205     qDebug() << Q_FUNC_INFO << "valide" << valide;
00206     return (valide && estValide());
00207 }
```

9.16.2.3 estValide()

```
IhmAuthentificationSansBadge::estValide ( )
```

Requete SQL pour autoriser l'accès de l'utilisateur.

Renvoie

```
valide
valide && estValide
```

Références [bdd](#), [dateValidite](#), [BaseDeDonnees : :recuperer\(\)](#), [ui](#), et [valide](#).

Référencé par [estAutorise\(\)](#).

```
00158 {
00159     QString dateMaintenant = QDateTime::currentDateTime().toString("yyyy-MM-dd");
00160     QString requete = "SELECT DateValidite FROM Utilisateur WHERE Identifiant='" +
ui->lineEditAuthentification->text() + "' AND DateValidite>='" + dateMaintenant + "'";
00161     qDebug() << Q_FUNC_INFO << "requete" << requete;
00162     QString retour;
00163     valide = bdd->recuperer(requete, retour);
00164     qDebug() << Q_FUNC_INFO << "valide" << valide << "dateValidite" <<
dateValidite;
00165     return valide;
00166 }
```

9.16.2.4 fermer

```
void IhmAuthentificationSansBadge::fermer ( ) [private], [slot]
```

Références [supprimerClavierVirtuel\(\)](#).

Référencé par [IhmAuthentificationSansBadge\(\)](#).

```
00122 {  
00123     supprimerClavierVirtuel();  
00124     close();  
00125 }
```

9.16.2.5 getIdUtilisateur()

```
QString IhmAuthentificationSansBadge::getIdUtilisateur ( ) const
```

Renvoie

idUtilisateur*

Références [idUtilisateur](#).

Référencé par [IhmAuthentification : :autoriserSansBadge\(\)](#).

```
00176 {  
00177     return idUtilisateur;  
00178 }
```

9.16.2.6 initialiserClavierVirtuel()

```
void IhmAuthentificationSansBadge::initialiserClavierVirtuel ( ) [private]
```

Référencé par [showEvent\(\)](#).

```
00094 {  
00095     QProcess processus;  
00096     QString programme = "/usr/bin/florence";  
00097     QStringList arguments;  
00098     arguments << "--use-config=/usr/share/doc/florence/examples/florence.conf";  
00099     processus.startDetached(programme, arguments);  
00100 }
```

9.16.2.7 initialiserFenetre()

```
void IhmAuthentificationSansBadge::initialiserFenetre ( ) [private]
```

Références [IHM_CSS](#).

Référencé par [IhmAuthentificationSansBadge\(\)](#).

```
00064 {
00065     setStyleSheet(IHM_CSS);
00066
00067     #ifndef DESKTOP
00068     const int w = qApp->desktop()->availableGeometry(this).width();
00069     const int h = qApp->desktop()->availableGeometry(this).height();
00070     resize(w, h);
00071     #else
00072     // 800x480
00073     resize(800, 480);
00074     #endif
00075     setWindowFlags(Qt::FramelessWindowHint | Qt::Dialog); // Ajouter Qt::WindowStaysOnTopHint
00076     //showMaximized();
00077     setWindowModality(Qt::WindowModal);
00078 }
```

9.16.2.8 showEvent()

```
void IhmAuthentificationSansBadge::showEvent (
    QShowEvent * ev ) [protected]
```

Références [initialiserClavierVirtuel\(\)](#), et [ui](#).

```
00081 {
00082     Q_UNUSED(ev)
00083
00084     initialiserClavierVirtuel();
00085     ui->lineEditAuthentification->setFocus();
00086 }
```

9.16.2.9 supprimerClavierVirtuel()

```
void IhmAuthentificationSansBadge::supprimerClavierVirtuel ( ) [private]
```

Référencé par [fermer\(\)](#).

```
00108 {
00109     QProcess processus;
00110     QString programme = "killall"; // le nom du programme
00111     QStringList arguments; // arguments du programme
00112     arguments << "florence";
00113     processus.startDetached(programme, arguments);
00114 }
```

9.16.2.10 utilisateurAutorise

```
void IhmAuthentificationSansBadge::utilisateurAutorise (
    QString identifiant,
    QString motDePasse ) [signal]
```

Référencé par [validerAcces\(\)](#).

9.16.2.11 validerAcces

```
void IhmAuthentificationSansBadge::validerAcces ( ) [private], [slot]
```

Références [effacerMessage\(\)](#), [estAutorise\(\)](#), [identifiant](#), [motDePasse](#), [ui](#), et [utilisateurAutorise\(\)](#).

Référencé par [IhmAuthentificationSansBadge\(\)](#).

```
00133 {
00134     if(estAutorise())
00135     {
00136         close();
00137         emit utilisateurAutorise(identifiant,
motDePasse);
00138     }
00139     else
00140     {
00141         ui->labelMessagAuthenticatioSansBadge->setText("Identifiant non valide !");
00142         QTimer::singleShot(1000, this, SLOT(effacerMessage()));
00143     }
00144     // Réinitialiser la saisie d'un nouvel utilisateur
00145     ui->lineEditAuthentification->setText("");
00146     ui->lineEditMotDePasse->setText("");
00147     ui->lineEditAuthentification->setFocus();
00148 }
```

9.16.3 Documentation des données membres

9.16.3.1 autorise

```
bool IhmAuthentificationSansBadge::autorise [private]
```

9.16.3.2 bdd

```
BaseDeDonnees* IhmAuthentificationSansBadge::bdd [private]
```

Référencé par [estAutorise\(\)](#), [estValide\(\)](#), et [IhmAuthentificationSansBadge\(\)](#).

9.16.3.3 dateValidite

```
QString IhmAuthentificationSansBadge::dateValidite [private]
```

Référencé par [estAutorise\(\)](#), et [estValide\(\)](#).

9.16.3.4 identifiant

```
QString IhmAuthentificationSansBadge::identifiant [private]
```

Référencé par [estAutorise\(\)](#), et [validerAcces\(\)](#).

9.16.3.5 idUtilisateur

```
QString IhmAuthentificationSansBadge::idUtilisateur [private]
```

Référencé par [estAutorise\(\)](#), et [getIdUtilisateur\(\)](#).

9.16.3.6 motDePasse

```
QString IhmAuthentificationSansBadge::motDePasse [private]
```

Référencé par [estAutorise\(\)](#), et [validerAcces\(\)](#).

9.16.3.7 ui

```
Ui::IhmAuthentificationSansBadge* IhmAuthentificationSansBadge::ui [private]
```

Référencé par [effacerMessage\(\)](#), [estAutorise\(\)](#), [estValide\(\)](#), [IhmAuthentificationSansBadge\(\)](#), [showEvent\(\)](#), [validerAcces\(\)](#), et [~IhmAuthentificationSansBadge\(\)](#).

9.16.3.8 valide

```
bool IhmAuthentificationSansBadge::valide [private]
```

Référencé par [estAutorise\(\)](#), et [estValide\(\)](#).

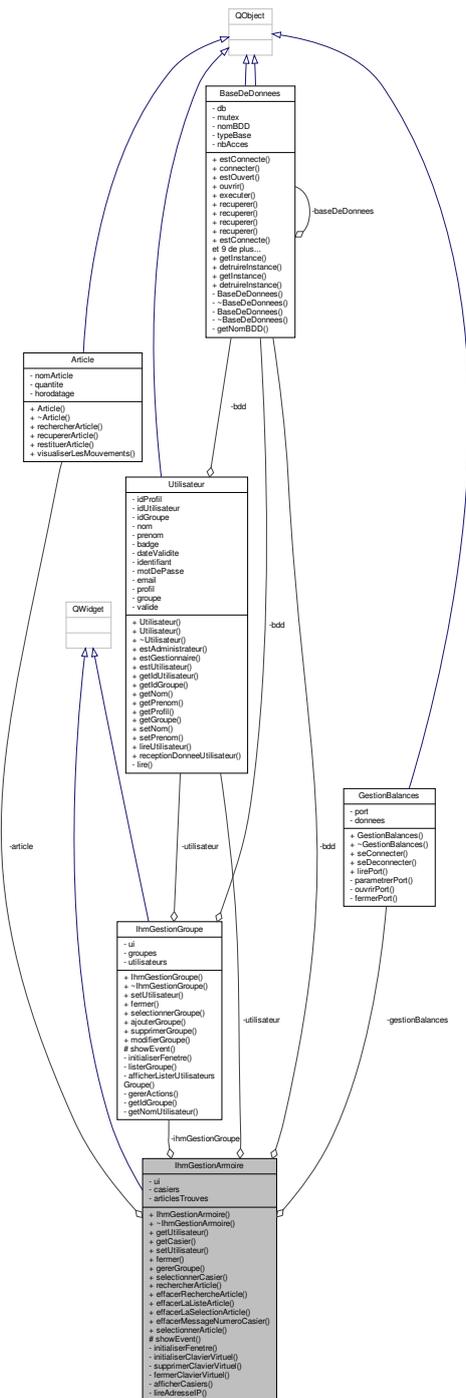
La documentation de cette classe a été générée à partir des fichiers suivants :

- [IhmAuthentificationSansBadge.h](#)
- [IhmAuthentificationSansBadge.cpp](#)
- [IhmGestionArmoire.cpp](#)

9.17 Référence de la classe IhmGestionArmoire

```
#include <IhmGestionArmoire.h>
```

Graphe de collaboration de IhmGestionArmoire :



Connecteurs publics

- void fermer ()
ferme l'IhmGestionArmoire et supprime l'utilisateur crée
- void gererGroupe ()
Identification du status de l'utilisateur gestionnaire ou administrateur.
- void selectionnerCasier (int)
Permet de savoir quel casier a été sélectionner.
- void rechercherArticle ()
Requete permettant la recherche d'article dans les casiers.
- void effacerRechercheArticle ()

- void `effacerLaListeArticle` ()
- void `effacerLaSelectionArticle` ()
- void `effacerMessageNumeroCasier` ()
- void `selectionnerArticle` (int index)

Fonctions membres publiques

- `IhmGestionArmoire` (QWidget *parent=nullptr)
- `~IhmGestionArmoire` ()
Destructeur de la fenêtre gestion armoire.
- `Utilisateur * getUtilisateur` ()
Retourne l'association utilisateur.
- `Casier * getCasier` ()
- void `setUtilisateur` (`Utilisateur *utilisateur`)
Crée l'association utilisateur.

Fonctions membres protégées

- void `showEvent` (QShowEvent *ev)
S'exécute à chaque fois que la fenêtre `IhmGestionArmoire` est affichée.

Fonctions membres privées

- void `initialiserFenetre` ()
Initialise le mode plein écran.
- void `initialiserClavierVirtuel` ()
Initialise le clavier virtuel.
- void `supprimerClavierVirtuel` ()
Supprime le clavier virtuel.
- void `fermerClavierVirtuel` ()
ferme le clavier virtuel
- void `afficherCasiers` ()
- QString `lireAdresseIP` ()

Attributs privés

- `Ui : :IhmGestionArmoire * ui`
agrégation de la définition de la fenêtre UI
- `BaseDeDonnees * bdd`
agrégation de l'objet `BaseDeDonnees`
- `Utilisateur * utilisateur`
- `GestionBalances * gestionBalances`
- `IhmGestionGroupe * ihmGestionGroupe`
la fenêtre de gestion des groupes
- `QVector< Casier * > casiers`
les casier de l'armoire
- `QVector< QStringList > articlesTrouves`
- `Article * article`

9.17.1 Documentation des constructeurs et destructeur

9.17.1.1 IhmGestionArmoire()

```
IhmGestionArmoire::IhmGestionArmoire (
    QWidget * parent = nullptr ) [explicit]
```

Références [afficherCasiers\(\)](#), [bdd](#), [BaseDeDonnees : :connecter\(\)](#), [BaseDeDonnees : :estConnecte\(\)](#), [fermer\(\)](#), [gererGroupe\(\)](#), [BaseDeDonnees : :getInstance\(\)](#), [ihmGestionGroupe](#), [initialiserFenetre\(\)](#), [rechercherArticle\(\)](#), [selectionnerArticle\(\)](#), et [ui](#).

```
00031                                     : QWidget(parent), ui(new Ui::IhmGestionArmoire),
    utilisateur(nullptr)//, casiers(NB_CASIER_MAX, nullptr)
00032 {
00033     ui->setupUi(this);
00034     qDebug() << Q_FUNC_INFO;
00035
00036     bdd = BaseDeDonnees::getInstance();
00037     if(!bdd->estConnecte())
00038         bdd->connecter();
00039
00040     // crée les casiers
00041     afficherCasiers();
00042
00043     ihmGestionGroupe = new IhmGestionGroupe(this);
00044     ihmGestionGroupe->hide();
00045
00046     // Temporaire
00047     //gestionBalances = new GestionBalances;
00048
00049     connect(ui->boutonFermeture, SIGNAL(clicked()), this, SLOT(fermer()));
00050     //connect(ui->lineEditRechercheArticle, SIGNAL(editingFinished()), this, SLOT(rechercherArticle()));
00051     connect(ui->boutonRechercheArticle, SIGNAL(clicked()), this, SLOT(
rechercherArticle()));
00052     connect(ui->comboBoxListeArticles, SIGNAL(currentIndexChanged(int)), this, SLOT(
selectionnerArticle(int));
00053     // pour administrateur
00054     connect(ui->boutonGestionGroupe, SIGNAL(clicked()), this, SLOT(gererGroupe()));
00055
00056     initialiserFenetre();
00057 }
```

9.17.1.2 ~IhmGestionArmoire()

```
IhmGestionArmoire::~IhmGestionArmoire ( )
```

Références [BaseDeDonnees : :destruireInstance\(\)](#), et [ui](#).

```
00066 {
00067     BaseDeDonnees::destruireInstance();
00068     delete ui;
00069     qDebug() << Q_FUNC_INFO;
00070 }
```

9.17.2 Documentation des fonctions membres

9.17.2.1 afficherCasiers()

```
void IhmGestionArmoire::afficherCasiers ( ) [private]
```

Références [bdd](#), [casiers](#), [lireAdresseIP\(\)](#), [BaseDeDonnees : :recuperer\(\)](#), [selectionnerCasier\(\)](#), et [ui](#).

Référencé par [IhmGestionArmoire\(\)](#).

```
00326 {
00327     QString nbCasiers;
00328     QString adresseIP = lireAdresseIP();
00329     qDebug() << Q_FUNC_INFO << "adresseIP" << adresseIP;
00330     if(!adresseIP.isEmpty())
00331     {
00332         QString requeteNbCasiers = "SELECT Armoire.nbCasiers FROM Armoire WHERE Armoire.AdresseIP = ' " +
adresseIP + "'";
00333         bdd->recuperer(requeteNbCasiers, nbCasiers);
00334         qDebug() << Q_FUNC_INFO << "nbCasiers" << nbCasiers;
00335         for(int i=0; i < nbCasiers.toInt(); i++)
00336         {
00337             Casier* casier = new Casier(i+1, true, false, this); // autorisé et présent
00338             casiers.push_back(casier);
00339             connect(casier, SIGNAL(clicked(int)), this, SLOT(selectionnerCasier(int)));
00340             // pair/impair -> droite/gauche ?
00341             if((i+1)%2)
00342             {
00343                 ui->gridLayoutArmoire->addWidget(casier, i/2, 0, 1, 1);
00344             }
00345             else
00346             {
00347                 ui->gridLayoutArmoire->addWidget(casier, i/2, 1, 1, 1);
00348             }
00349         }
00350     }
00351 }
```

9.17.2.2 effacerLaListeArticle

```
void IhmGestionArmoire::effacerLaListeArticle ( ) [slot]
```

Références [ui](#).

Référencé par [fermer\(\)](#).

```
00311 {
00312     ui->comboBoxListeArticles->clear();
00313 }
```

9.17.2.3 effacerLaSelectionArticle

```
void IhmGestionArmoire::effacerLaSelectionArticle ( ) [slot]
```

Références [ui](#).

Référencé par [fermer\(\)](#).

```
00316 {
00317     ui->labelMessageArticle->setText("");
00318 }
```

9.17.2.4 effacerMessageNumeroCasier

```
void IhmGestionArmoire::effacerMessageNumeroCasier ( ) [slot]
```

Références [ui](#).

Référencé par [fermer\(\)](#).

```
00321 {  
00322     ui->labelMessageNumeroCasier->setText("");  
00323 }
```

9.17.2.5 effacerRechercheArticle

```
void IhmGestionArmoire::effacerRechercheArticle ( ) [slot]
```

Références [ui](#).

Référencé par [rechercherArticle\(\)](#).

```
00280 {  
00281     ui->lineEditRechercheArticle->setText("");  
00282 }
```

9.17.2.6 fermer

```
void IhmGestionArmoire::fermer ( ) [slot]
```

Références [effacerLaListeArticle\(\)](#), [effacerLaSelectionArticle\(\)](#), [effacerMessageNumeroCasier\(\)](#), et [utilisateur](#).

Référencé par [IhmGestionArmoire\(\)](#).

```
00195 {  
00196     if (utilisateur != nullptr)  
00197     {  
00198         delete utilisateur;  
00199         utilisateur = nullptr;  
00200     }  
00201     close();  
00202     effacerLaListeArticle();  
00203     effacerLaSelectionArticle();  
00204     effacerMessageNumeroCasier();  
00205 }
```

9.17.2.7 fermerClavierVirtuel()

```
void IhmGestionArmoire::fermerClavierVirtuel ( ) [private]
```

Références [supprimerClavierVirtuel\(\)](#).

```
00183 {  
00184     supprimerClavierVirtuel();  
00185     close();  
00186 }
```

9.17.2.8 gererGroupe

```
void IhmGestionArmoire::gererGroupe ( ) [slot]
```

Références [Utilisateur : :estAdministrateur\(\)](#), [Utilisateur : :estGestionnaire\(\)](#), [Utilisateur : :getNom\(\)](#), [ihmGestionGroupe](#), [IhmGestionGroupe : :setUtilisateur\(\)](#), et [utilisateur](#).

Référencé par [IhmGestionArmoire\(\)](#).

```
00215 {
00216     qDebug() << Q_FUNC_INFO << "Nom" << utilisateur->getNom() << "Administrateur" <<
        utilisateur->estAdministrateur() << "estGestionnaire" <<
        utilisateur->estGestionnaire();
00217     if (utilisateur->estAdministrateur() ||
        utilisateur->estGestionnaire())
00218     {
00219         ihmGestionGroupe->setUtilisateur(
        utilisateur);
00220         ihmGestionGroupe->show();
00221     }
00222 }
```

9.17.2.9 getCasier()

```
Casier* IhmGestionArmoire::getCasier ( )
```

9.17.2.10 getUtilisateur()

```
Utilisateur * IhmGestionArmoire::getUtilisateur ( )
```

Renvoie

Utilisateur*

Références [utilisateur](#).

```
00080 {
00081     return utilisateur;
00082 }
```

9.17.2.11 initialiserClavierVirtuel()

```
void IhmGestionArmoire::initialiserClavierVirtuel ( ) [private]
```

Référencé par [showEvent\(\)](#).

```
00153 {
00154     QProcess processus;
00155     QString programme = "/usr/bin/florence";
00156     QStringList arguments;
00157     arguments << "--use-config=/usr/share/doc/florence/examples/florence.conf";
00158     processus.startDetached(programme, arguments);
00159 }
```

9.17.2.12 initialiserFenetre()

```
void IhmGestionArmoire::initialiserFenetre ( ) [private]
```

Références [IHM_CSS](#), et [ui](#).

Référencé par [IhmGestionArmoire\(\)](#).

```
00129 {
00130     setStyleSheet (IHM_CSS);
00131
00132     #ifndef DESKTOP
00133     const int w = qApp->desktop()->availableGeometry(this).width();
00134     const int h = qApp->desktop()->availableGeometry(this).height();
00135     resize(w, h);
00136     #else
00137     // 800x480
00138     resize(800, 480);
00139     #endif
00140     setWindowFlags (Qt::FramelessWindowHint | Qt::Dialog); // Ajouter Qt::WindowStaysOnTopHint
00141     //showMaximized();
00142     setWindowModality (Qt::WindowModal);
00143     ui->lineEditRechercheArticle->setFocus();
00144 }
```

9.17.2.13 lireAdresseIP()

```
QString IhmGestionArmoire::lireAdresseIP ( ) [private]
```

Référencé par [afficherCasiers\(\)](#).

```
00354 {
00355     //QNetworkInterface interface = QNetworkInterface::interfaceFromName("wlan0");
00356     //QDebug() << Q_FUNC_INFO << interface.hardwareAddress();
00357     QList<QHostAddress> addresses = QNetworkInterface::allAddresses();
00358     for(int i = 0; addresses.size(); i++)
00359     {
00360         if(addresses.at(i).isInSubnet(QHostAddress::parseSubnet("192.168.52.0/24")))
00361             return addresses.at(i).toString();
00362     }
00363     return QString("");
00364 }
00365 }
```

9.17.2.14 rechercherArticle

```
IhmGestionArmoire::rechercherArticle ( ) [slot]
```

Affiche l'article selectionner.

Références [articlesTrouves](#), [bdd](#), [effacerRechercheArticle\(\)](#), [BaseDeDonnees : :recuperer\(\)](#), [selectionnerArticle\(\)](#), et [ui](#).

Référencé par [IhmGestionArmoire\(\)](#).

```

00250 {
00251     QString mot = ui->lineEditRechercheArticle->text();
00252     qDebug() << Q_FUNC_INFO << mot;
00253     /*if(mot.isEmpty())
00254         return;*/
00255     QString requete = "SELECT Stock.NumeroCasier, Article.idType, Article.Nom, Stock.Quantite,
Stock.Disponible, Article.Designation FROM Stock INNER JOIN Article ON Stock.idArticle = Article.idArticle WHERE
Article.Nom LIKE '%" + mot + "%' OR Article.Code LIKE '%" + mot + "%' OR Article.Designation LIKE '%" + mot + "%'
ORDER BY Stock.NumeroCasier ASC";
00256     //QString requete = "SELECT * FROM Article WHERE Nom LIKE '%" + mot + "%' OR Code LIKE '%" + mot + "%'
OR Designation LIKE '%" + mot + "%'";
00257     qDebug() << Q_FUNC_INFO << requete;
00258     disconnect(ui->comboBoxListeArticles, SIGNAL(currentIndexChanged(int)), this, SLOT(
selectionnerArticle(int)));
00259     ui->comboBoxListeArticles->clear();
00260     articlesTrouves.clear();
00261     bdd->recuperer(requete, articlesTrouves);
00262     qDebug() << Q_FUNC_INFO << "articlesTrouves" << articlesTrouves.size() <<
articlesTrouves;
00263
00264     ui->comboBoxListeArticles->addItem("Sélectionner un article");
00265     for(int i = 0 ; i < articlesTrouves.size() ; i++)
00266     {
00267         ui->comboBoxListeArticles->addItem(articlesTrouves[i].at(2));
00268     }
00269     connect(ui->comboBoxListeArticles, SIGNAL(currentIndexChanged(int)), this, SLOT(
selectionnerArticle(int)));
00270
00271     /*if(articlesTrouves.size() > 0)
00272     {
00273         selectionnerArticle(1);
00274         ui->labelMessageNumeroCasier->setText("L'article se trouve dans le casier numéro : " +
articlesTrouves[0].at(0));
00275     }*/
00276     effacerRechercheArticle();
00277 }

```

9.17.2.15 selectionnerArticle

```

void IhmGestionArmoire::selectionnerArticle (
    int index ) [slot]

```

Références [article](#), [articlesTrouves](#), et [ui](#).

Référencé par [IhmGestionArmoire\(\)](#), et [rechercherArticle\(\)](#).

```

00291 {
00292     if(articlesTrouves.size() == 0 || index < 0)
00293         return;
00294     if(index > 0)
00295     {
00296         //Stock.NumeroCasier, Article.idType, Article.Nom, Stock.Quantite, Stock.Disponible,
Article.Designation
00297         QString article = articlesTrouves[index-1].at(5);
00298         QString qte = QString::fromUtf8("Quantité : ") + articlesTrouves[index-1].at(3);
00299         QString dispo = QString::fromUtf8("Disponible : ") + articlesTrouves[index-1].at(4);
00300         ui->labelMessageArticle->setText(article + "\n" + qte + "\n" + dispo);
00301         ui->labelMessageNumeroCasier->setText("Casier " + articlesTrouves[index-1].at(0));
00302     }
00303     else
00304     {
00305         ui->labelMessageArticle->setText("");
00306         ui->labelMessageNumeroCasier->setText("");
00307     }
00308 }

```

9.17.2.16 selectionnerCasier

```

void IhmGestionArmoire::selectionnerCasier (
    int numeroCasier ) [slot]

```

Paramètres

<i>numeroCasier</i>	int
---------------------	-----

Références [bdd](#), [casiers](#), et [BaseDeDonnees : :recuperer\(\)](#).

Référencé par [afficherCasiers\(\)](#).

```
00233 {
00234     qDebug() << Q_FUNC_INFO << "numeroCasier" << numeroCasier;
00235     QString requete = "SELECT Stock.NumeroCasier, Article.idType, Article.Nom, Stock.Quantite,
    Stock.Disponible, Article.Designation FROM Stock INNER JOIN Article ON Stock.idArticle = Article.idArticle WHERE
    Stock.NumeroCasier=' " + QString::number(numeroCasier) + "'";
00236     qDebug() << Q_FUNC_INFO << requete;
00237     QStringList articleCasier;
00238     bdd->recuperer(requete, articleCasier);
00239     qDebug() << Q_FUNC_INFO << "articleCasier" << articleCasier;
00240     casiers[numeroCasier-1]->actionner();
00241 }
```

9.17.2.17 setUtilisateur()

```
IhmGestionArmoire::setUtilisateur (
    Utilisateur * utilisateur )
```

Paramètres

<i>utilisateur</i>	Utilisateur*
--------------------	--------------

Références [Utilisateur : :estAdministrateur\(\)](#), [Utilisateur : :estGestionnaire\(\)](#), [ui](#), et [utilisateur](#).

Référencé par [IhmAuthentification : :autoriserSansBadge\(\)](#), [IhmAuthentification : :passerSansBadge\(\)](#), et [IhmAuthentification : :validerAcces\(\)](#).

```
00092 {
00093     this->utilisateur = utilisateur;
00094     if(utilisateur->estAdministrateur() || utilisateur->
estGestionnaire())
00095         ui->boutonGestionGroupe->setEnabled(true);
00096     else
00097         ui->boutonGestionGroupe->setEnabled(false);
00098 }
```

9.17.2.18 showEvent()

```
void IhmGestionArmoire::showEvent (
    QShowEvent * ev ) [protected]
```

Références [Utilisateur : :getGroupe\(\)](#), [Utilisateur : :getNom\(\)](#), [Utilisateur : :getPrenom\(\)](#), [initialiserClavierVirtuel\(\)](#), [ui](#), et [utilisateur](#).

```
00107 {
00108     QString informationsUtilisateur;
00109     QString informationsGroupe;
00110     // Affiche des informations sur l'utilisateur qui a accès à l'armoire
00111     if(utilisateur != nullptr)
00112     {
00113         informationsUtilisateur = utilisateur->getNom() + " " +
utilisateur->getPrenom();
00114         informationsGroupe = utilisateur->getGroupe();
00115     }
00116     ui->labelUtilisateur->setText(informationsUtilisateur);
00117     ui->labelGroupe->setText(informationsGroupe);
00118
00119     initialiserClavierVirtuel();
00120 }
```

9.17.2.19 supprimerClavierVirtuel()

```
void IhmGestionArmoire::supprimerClavierVirtuel ( ) [private]
```

Référencé par [fermerClavierVirtuel\(\)](#).

```
00168 {  
00169     QProcess processus;  
00170     QString programme = "killall"; // le nom du programme  
00171     QStringList arguments; // arguments du programme  
00172     arguments << "florence";  
00173     processus.startDetached(programme, arguments);  
00174 }
```

9.17.3 Documentation des données membres

9.17.3.1 article

```
Article* IhmGestionArmoire::article [private]
```

Référencé par [selectionnerArticle\(\)](#).

9.17.3.2 articlesTrouves

```
QVector<QStringList> IhmGestionArmoire::articlesTrouves [private]
```

Référencé par [rechercherArticle\(\)](#), et [selectionnerArticle\(\)](#).

9.17.3.3 bdd

```
BaseDeDonnees* IhmGestionArmoire::bdd [private]
```

Référencé par [afficherCasiers\(\)](#), [IhmGestionArmoire\(\)](#), [rechercherArticle\(\)](#), et [selectionnerCasier\(\)](#).

9.17.3.4 casiers

```
QVector<Casier *> IhmGestionArmoire::casiers [private]
```

Référencé par [afficherCasiers\(\)](#), et [selectionnerCasier\(\)](#).

9.17.3.5 gestionBalances

```
GestionBalances* IhmGestionArmoire::gestionBalances [private]
```

9.17.3.6 ihmGestionGroupe

```
IhmGestionGroupe* IhmGestionArmoire::ihmGestionGroupe [private]
```

Référencé par [gererGroupe\(\)](#), et [IhmGestionArmoire\(\)](#).

9.17.3.7 ui

```
Ui::IhmGestionArmoire* IhmGestionArmoire::ui [private]
```

Référencé par [afficherCasiers\(\)](#), [effacerLaListeArticle\(\)](#), [effacerLaSelectionArticle\(\)](#), [effacerMessageNumeroCasier\(\)](#), [effacerRechercheArticle\(\)](#), [IhmGestionArmoire\(\)](#), [initialiserFenetre\(\)](#), [rechercherArticle\(\)](#), [selectionnerArticle\(\)](#), [setUtilisateur\(\)](#), [showEvent\(\)](#), et [~IhmGestionArmoire\(\)](#).

9.17.3.8 utilisateur

```
Utilisateur* IhmGestionArmoire::utilisateur [private]
```

Référencé par [fermer\(\)](#), [gererGroupe\(\)](#), [getUtilisateur\(\)](#), [setUtilisateur\(\)](#), et [showEvent\(\)](#).

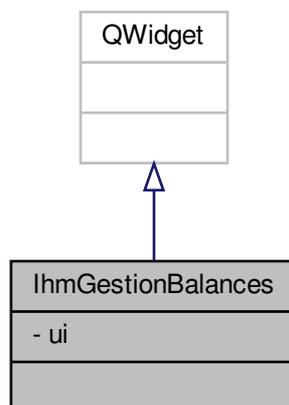
La documentation de cette classe a été générée à partir des fichiers suivants :

- [IhmGestionArmoire.h](#)
- [IhmGestionArmoire.cpp](#)
- [IhmGestionGroupe.cpp](#)

9.18 Référence de la classe IhmGestionBalances

```
#include <IhmGestionBalances.h>
```

Graphe de collaboration de IhmGestionBalances :



Attributs privés

— `Ui : :IhmGestionBalances * ui`

9.18.1 Documentation des données membres

9.18.1.1 `ui`

```
Ui::IhmGestionBalances* IhmGestionBalances::ui [private]
```

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

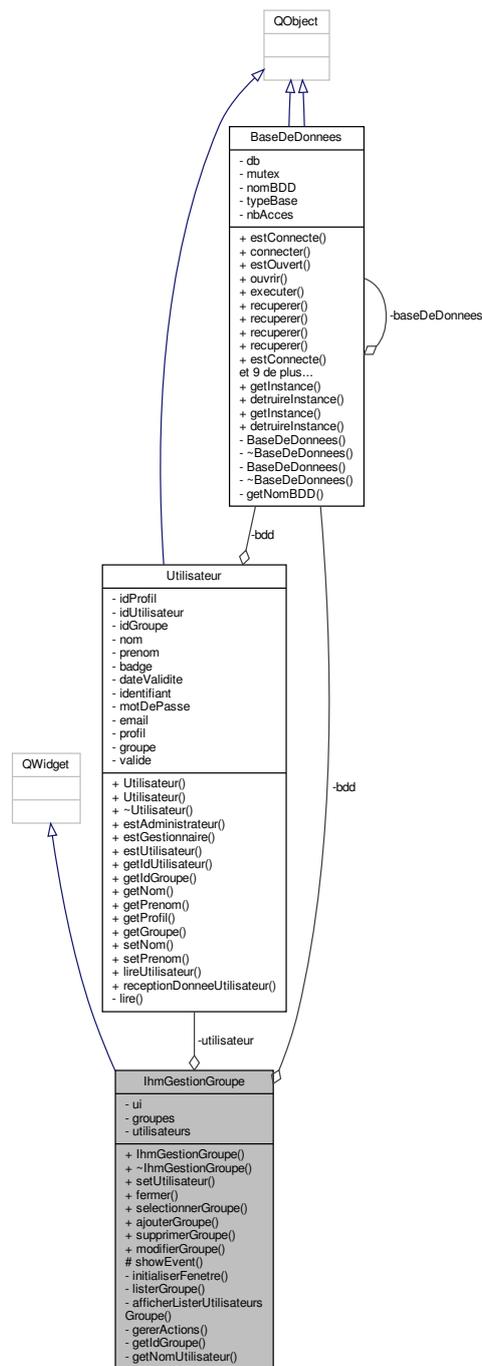
— [IhmGestionBalances.h](#)

9.19 Référence de la classe `IhmGestionGroupe`

La fenêtre de gestion des groupes.

```
#include <IhmGestionGroupe.h>
```

Grphe de collaboration de IhmGestionGroupe :



Connecteurs publics

- void **fermer** ()
Ferme la fenêtre.
- void **selectionnerGroupe** (QString nomGroupe)
Sélectionne un groupe dans la liste.
- void **ajouterGroupe** ()
Ajouter un groupe.
- void **supprimerGroupe** ()
Supprimer un groupe.
- void **modifierGroupe** ()
Modifier un groupe.

Fonctions membres publiques

- `IhmGestionGroupe` (QWidget *parent=nullptr)
Constructeur de la Gestion de groupe.
- `~IhmGestionGroupe` ()
Destructeur de la fenêtre Gestion de Groupe.
- void `setUtilisateur` (Utilisateur *utilisateur)

Fonctions membres protégées

- void `showEvent` (QShowEvent *ev)
Recharge la liste des groupes à chaque fois que la fenêtre GestionGroupe est affichée.

Fonctions membres privées

- void `initialiserFenetre` ()
Initialise le mode plein écran.
- void `listerGroupe` ()
Affiche la liste des groupes.
- void `afficherListerUtilisateursGroupe` ()
Affiche la liste des utilisateur pour le groupe sélectionné
- void `gererActions` (bool etat)
Gère les état des boutons Ajout, Modification et Suppression.
- QString `getIdGroupe` ()
Retourne l'idGroupe du groupe sélectionné
- QString `getNomUtilisateur` ()

Attributs privés

- Ui : :IhmGestionGroupe * `ui`
agrégation de la définition de la fenêtre UI
- `BaseDeDonnees` * `bdd`
agrégation de l'objet BaseDeDonnees
- `Utilisateur` * `utilisateur`
- QVector< QStringList > `groupes`
les groupes
- QStringList `utilisateurs`

9.19.1 Description détaillée

Auteur

Hadrien GIMENEZ

Version

1.1

9.19.2 Documentation des constructeurs et destructeur

9.19.2.1 IhmGestionGroupe()

```
IhmGestionGroupe::IhmGestionGroupe (
    QWidget * parent = nullptr ) [explicit]
```

Paramètres

<i>parent</i>	QWidget*
---------------	----------

Références [ajouterGroupe\(\)](#), [bdd](#), [BaseDeDonnees : :connecter\(\)](#), [BaseDeDonnees : :estConnecte\(\)](#), [fermer\(\)](#), [BaseDeDonnees : :getInstance\(\)](#), [initialiserFenetre\(\)](#), [modifierGroupe\(\)](#), [supprimerGroupe\(\)](#), et [ui](#).

```

00013         :
00014     QWidget (parent),
00015     ui (new Ui::IhmGestionGroupe), utilisateur(nullptr)
00016 {
00017     ui->setupUi (this);
00018     qDebug() << Q_FUNC_INFO;
00019
00020     bdd = BaseDeDonnees::getInstance();
00021     if (!bdd->estConnecte())
00022         bdd->connecter();
00023
00024     connect (ui->boutonFermeture, SIGNAL(clicked()), this, SLOT(fermer()));
00025     connect (ui->boutonAjout, SIGNAL(clicked()), this, SLOT(ajouterGroupe()));
00026     connect (ui->boutonSuppression, SIGNAL(clicked()), this, SLOT(
supprimerGroupe()));
00027     connect (ui->boutonModification, SIGNAL(clicked()), this, SLOT(
modifierGroupe()));
00028
00029     ui->boutonAjout->setEnabled(false);
00030     ui->boutonModification->setEnabled(false);
00031     ui->boutonSuppression->setEnabled(false);
00032     QStringListModel *modele = new QStringListModel(this);
00033     modele->setStringList(QStringList{});
00034     ui->listViewUtilisateurs->setModel(modele);
00035
00036     initialiserFenetre();
00037 }

```

9.19.2.2 ~IhmGestionGroupe()

```
IhmGestionGroupe::~IhmGestionGroupe ( )
```

Références [BaseDeDonnees : :destruireInstance\(\)](#), et [ui](#).

```

00046 {
00047     BaseDeDonnees::destruireInstance();
00048     delete ui;
00049     qDebug() << Q_FUNC_INFO;
00050 }

```

9.19.3 Documentation des fonctions membres

9.19.3.1 afficherListerUtilisateursGroupe()

```
void IhmGestionGroupe::afficherListerUtilisateursGroupe ( ) [private]
```

Références [ui](#), et [utilisateurs](#).

Référencé par [selectionnerGroupe\(\)](#).

```

00157 {
00158     QStringListModel *modele;
00159     modele = new QStringListModel(this);
00160     modele->setStringList(utilisateurs);
00161     ui->listViewUtilisateurs->setModel(modele);
00162 }

```

9.19.3.2 ajouterGroupe

```
void IhmGestionGroupe::ajouterGroupe ( ) [slot]
```

Références [bdd](#), [Utilisateur : :estAdministrateur\(\)](#), [Utilisateur : :estGestionnaire\(\)](#), [BaseDeDonnees : :executer\(\)](#), [Utilisateur : :get←Nom\(\)](#), [listerGroupe\(\)](#), [ui](#), et [utilisateur](#).

Référencé par [IhmGestionGroupe\(\)](#).

```
00221 {
00222     if(utilisateur != nullptr)
00223     {
00224         qDebug() << Q_FUNC_INFO << "Nom" << utilisateur->getNom() << "Administrateur" <<
utilisateur->estAdministrateur() << "estGestionnaire" <<
utilisateur->estGestionnaire();
00225         if(utilisateur->estAdministrateur() ||
utilisateur->estGestionnaire())
00226         {
00227             QString nomGroupe = ui->saisieNomGroupe->text();
00228             if(nomGroupe.isEmpty())
00229                 return;
00230             QString requete = QString("INSERT INTO 'Groupe' ('Nom') VALUES('%1')").arg(nomGroupe);
00231             qDebug() << Q_FUNC_INFO << "requete" << requete;
00232             bool retour = bdd->executer(requete);
00233
00234             if(retour)
00235             {
00236                 ui->saisieNomGroupe->clear();
00237                 listerGroupe();
00238             }
00239         }
00240     }
00241 }
```

9.19.3.3 fermer

```
void IhmGestionGroupe::fermer ( ) [slot]
```

Références [ui](#).

Référencé par [IhmGestionGroupe\(\)](#).

```
00106 {
00107     ui->saisieNomGroupe->setText("");
00108     ui->cbListeGroupe->setCurrentIndex(0);
00109     ui->boutonAjout->setEnabled(false);
00110     ui->boutonModification->setEnabled(false);
00111     ui->boutonSuppression->setEnabled(false);
00112     close();
00113 }
```

9.19.3.4 gererActions()

```
void IhmGestionGroupe::gererActions (
    bool etat ) [private]
```

Paramètres

<i>etat</i>	bool true (Modification et Suppression actifs et pas d'AJout) sinon l'inverse
-------------	---

Références [ui](#).

Référencé par [selectionnerGroupe\(\)](#), et [showEvent\(\)](#).

```
00172 {
00173     ui->boutonModification->setEnabled(etat);
00174     ui->boutonSuppression->setEnabled(etat);
00175     ui->boutonAjout->setEnabled(!etat);
00176 }
```

9.19.3.5 getIdGroupe()

QString IhmGestionGroupe::getIdGroupe () [private]

Renvoie

QString l'idGroupe

Références [groupes](#), et [ui](#).

Référencé par [modifierGroupe\(\)](#), [selectionnerGroupe\(\)](#), et [supprimerGroupe\(\)](#).

```
00316 {
00317     QString idGroupe = "";
00318     QString nomGroupe = ui->cbListeGroupe->currentText();
00319     if(nomGroupe.isEmpty())
00320         return idGroupe;
00321
00322     for(int i=0;i<groupes.size();i++)
00323     {
00324         if(groupes.at(i).at(1) == nomGroupe)
00325         {
00326             idGroupe = groupes.at(i).at(0);
00327             break;
00328         }
00329     }
00330
00331     qDebug() << Q_FUNC_INFO << "idGroupe" << idGroupe;
00332     return idGroupe;
00333 }
```

9.19.3.6 getNomUtilisateur()

QString IhmGestionGroupe::getNomUtilisateur () [private]

```
00336 {
00337     return QString("");
00338 }
```

9.19.3.7 initialiserFenetre()

void IhmGestionGroupe::initialiserFenetre () [private]

Références [IHM_CSS](#).

Référencé par [IhmGestionGroupe\(\)](#).

```
00071 {
00072     setStyleSheet(IHM_CSS);
00073
00074     #ifndef DESKTOP
00075     const int w = qApp->desktop()->availableGeometry(this).width();
00076     const int h = qApp->desktop()->availableGeometry(this).height();
00077     resize(w, h);
00078     #else
00079     // 800x480
00080     resize(800, 480);
00081     #endif
00082     setWindowFlags(Qt::FramelessWindowHint | Qt::Dialog); // Ajouter Qt::WindowStaysOnTopHint
00083     //showMaximized();
00084     setWindowModality(Qt::WindowModal);
00085 }
```

9.19.3.8 listerGroupe()

```
void IhmGestionGroupe::listerGroupe ( ) [private]
```

Références [bdd](#), [Utilisateur : :estAdministrateur\(\)](#), [Utilisateur : :estGestionnaire\(\)](#), [Utilisateur : :getNom\(\)](#), [groupes](#), [BaseDeDonnees : :recuperer\(\)](#), [selectionnerGroupe\(\)](#), [ui](#), et [utilisateur](#).

Référencé par [ajouterGroupe\(\)](#), [modifierGroupe\(\)](#), [showEvent\(\)](#), et [supprimerGroupe\(\)](#).

```
00185 {
00186     QString requete;
00187
00188     if(utilisateur != nullptr)
00189     {
00190         qDebug() << Q_FUNC_INFO << "Nom" << utilisateur->getNom() << "Administrateur" <<
utilisateur->estAdministrateur() << "estGestionnaire" <<
utilisateur->estGestionnaire();
00191         if(utilisateur->estAdministrateur() ||
utilisateur->estGestionnaire())
00192         {
00193             requete = "SELECT * FROM Groupe";
00194             groupes.clear();
00195             bool retour = bdd->recuperer(requete, groupes);
00196             if(retour)
00197             {
00198                 qDebug() << Q_FUNC_INFO << groupes;
00199                 disconnect(ui->cbListeGroupe, SIGNAL(currentIndexChanged(QString)), this, SLOT(
selectionnerGroupe(QString)));
00200                 if(ui->listViewUtilisateurs->model()->rowCount() > 0)
00201                     ui->listViewUtilisateurs->model()->removeRows(0, ui->listViewUtilisateurs->model()
->rowCount());
00202                 ui->cbListeGroupe->clear();
00203                 ui->cbListeGroupe->addItem("");
00204                 for(int i=0;i<groupes.size();i++)
00205                 {
00206                     ui->cbListeGroupe->addItem(groupes.at(i).at(1));
00207                 }
00208                 connect(ui->cbListeGroupe, SIGNAL(currentIndexChanged(QString)), this, SLOT(
selectionnerGroupe(QString)));
00209             }
00210         }
00211     }
00212 }
```

9.19.3.9 modifierGroupe

```
void IhmGestionGroupe::modifierGroupe ( ) [slot]
```

Références [bdd](#), [Utilisateur : :estAdministrateur\(\)](#), [Utilisateur : :estGestionnaire\(\)](#), [BaseDeDonnees : :executer\(\)](#), [getIdGroupe\(\)](#), [Utilisateur : :getNom\(\)](#), [listerGroupe\(\)](#), [ui](#), et [utilisateur](#).

Référencé par [IhmGestionGroupe\(\)](#).

```
00283 {
00284     if(utilisateur != nullptr)
00285     {
00286         qDebug() << Q_FUNC_INFO << "Nom" << utilisateur->getNom() << "Administrateur" <<
utilisateur->estAdministrateur() << "estGestionnaire" <<
utilisateur->estGestionnaire();
00287         if(utilisateur->estAdministrateur() ||
utilisateur->estGestionnaire())
00288         {
00289             QString nomGroupe = ui->saisieNomGroupe->text();
00290             if(nomGroupe.isEmpty())
00291                 return;
00292             QString idGroupe = getIdGroupe();
00293             if(idGroupe.isEmpty())
00294                 return;
00295
00296             QString requete = QString("UPDATE `Groupe` SET `Nom`='%1' WHERE idGroupe='%2'").arg(nomGroupe).
arg(idGroupe);
00297             qDebug() << Q_FUNC_INFO << "requete" << requete;
00298             bool retour = bdd->executer(requete);
00299
00300             if(retour)
00301             {
00302                 ui->saisieNomGroupe->clear();
00303                 listerGroupe();
00304             }
00305         }
00306     }
00307 }
```

9.19.3.10 selectionnerGroupe

```
void IhmGestionGroupe::selectionnerGroupe (
    QString nomGroupe ) [slot]
```

Paramètres

<i>nomGroupe</i>	QString le nom du groupe sélectionné
------------------	--------------------------------------

Références [afficherListerUtilisateursGroupe\(\)](#), [bdd](#), [gererActions\(\)](#), [getIdGroupe\(\)](#), [BaseDeDonnees : :recuperer\(\)](#), [ui](#), et [utilisateurs](#).

Référencé par [listerGroupe\(\)](#).

```
00122 {
00123     if (!nomGroupe.isEmpty())
00124     {
00125         ui->saisieNomGroupe->setText(nomGroupe);
00126         QVector<QStringList> donnees;
00127         QString requete = "SELECT Nom, Prenom FROM Utilisateur WHERE idGroupe=" +
    getIdGroupe() + "'";
00128         bool retour = bdd->recuperer(requete, donnees);
00129         if (retour)
00130         {
00131             qDebug() << Q_FUNC_INFO << donnees;
00132             utilisateurs.clear();
00133             for (int i = 0; i < donnees.size(); i++)
00134             {
00135                 utilisateurs << (donnees.at(i).at(0) + " " + donnees.at(i).at(1));
00136             }
00137             afficherListerUtilisateursGroupe();
00138             gererActions(true);
00139         }
00140     }
00141     else
00142     {
00143         ui->saisieNomGroupe->setText("");
00144         if (ui->listViewUtilisateurs->model()->rowCount() > 0)
00145             ui->listViewUtilisateurs->model()->removeRows(0, ui->listViewUtilisateurs->model()->
    rowCount());
00146         gererActions(false);
00147     }
00148 }
```

9.19.3.11 setUtilisateur()

```
void IhmGestionGroupe::setUtilisateur (
    Utilisateur * utilisateur )
```

Références [utilisateur](#).

Référencé par [IhmGestionArmoire : :gererGroupe\(\)](#).

```
00060 {
00061     this->utilisateur = utilisateur;
00062 }
```

9.19.3.12 showEvent()

```
void IhmGestionGroupe::showEvent (
    QShowEvent * ev ) [protected]
```

Références [gererActions\(\)](#), et [listerGroupe\(\)](#).

```
00094 {
00095     listerGroupe();
00096     gererActions(false);
00097 }
```

9.19.3.13 supprimerGroupe

```
void IhmGestionGroupe::supprimerGroupe ( ) [slot]
```

Références [bdd](#), [Utilisateur : :estAdministrateur\(\)](#), [Utilisateur : :estGestionnaire\(\)](#), [BaseDeDonnees : :executer\(\)](#), [getIdGroupe\(\)](#), [Utilisateur : :getNom\(\)](#), [listerGroupe\(\)](#), [ui](#), et [utilisateur](#).

Référencé par [IhmGestionGroupe\(\)](#).

```
00251 {
00252     if(utilisateur != nullptr)
00253     {
00254         qDebug() << Q_FUNC_INFO << "Nom" << utilisateur->getNom() << "Administrateur" <<
utilisateur->estAdministrateur() << "estGestionnaire" <<
utilisateur->estGestionnaire();
00255         if(utilisateur->estAdministrateur() ||
utilisateur->estGestionnaire())
00256         {
00257             QString nomGroupe = ui->saisieNomGroupe->text();
00258             if(nomGroupe.isEmpty())
00259                 return;
00260             QString idGroupe = getIdGroupe();
00261             if(idGroupe.isEmpty())
00262                 return;
00263             QString requete = QString("DELETE FROM `Groupe` WHERE idGroupe='%1'").arg(idGroupe);
00264             qDebug() << Q_FUNC_INFO << "requete" << requete;
00265             bool retour = bdd->executer(requete);
00266
00267             if(retour)
00268             {
00269                 ui->saisieNomGroupe->clear();
00270                 listerGroupe();
00271             }
00272         }
00273     }
00274 }
```

9.19.4 Documentation des données membres

9.19.4.1 bdd

```
BaseDeDonnees* IhmGestionGroupe::bdd [private]
```

Référencé par [ajouterGroupe\(\)](#), [IhmGestionGroupe\(\)](#), [listerGroupe\(\)](#), [modifierGroupe\(\)](#), [selectionnerGroupe\(\)](#), et [supprimerGroupe\(\)](#).

9.19.4.2 groupes

```
QVector<QStringList> IhmGestionGroupe::groupes [private]
```

Référencé par [getIdGroupe\(\)](#), et [listerGroupe\(\)](#).

9.19.4.3 ui

```
Ui::IhmGestionGroupe* IhmGestionGroupe::ui [private]
```

Référencé par [afficherListerUtilisateursGroupe\(\)](#), [ajouterGroupe\(\)](#), [fermer\(\)](#), [gererActions\(\)](#), [getIdGroupe\(\)](#), [IhmGestionGroupe\(\)](#), [listerGroupe\(\)](#), [modifierGroupe\(\)](#), [selectionnerGroupe\(\)](#), [supprimerGroupe\(\)](#), et [~IhmGestionGroupe\(\)](#).

9.19.4.4 utilisateur

```
Utilisateur* IhmGestionGroupe::utilisateur [private]
```

Référencé par [ajouterGroupe\(\)](#), [listerGroupe\(\)](#), [modifierGroupe\(\)](#), [setUtilisateur\(\)](#), et [supprimerGroupe\(\)](#).

9.19.4.5 utilisateurs

```
QStringList IhmGestionGroupe::utilisateurs [private]
```

Référencé par [afficherListerUtilisateursGroupe\(\)](#), et [selectionnerGroupe\(\)](#).

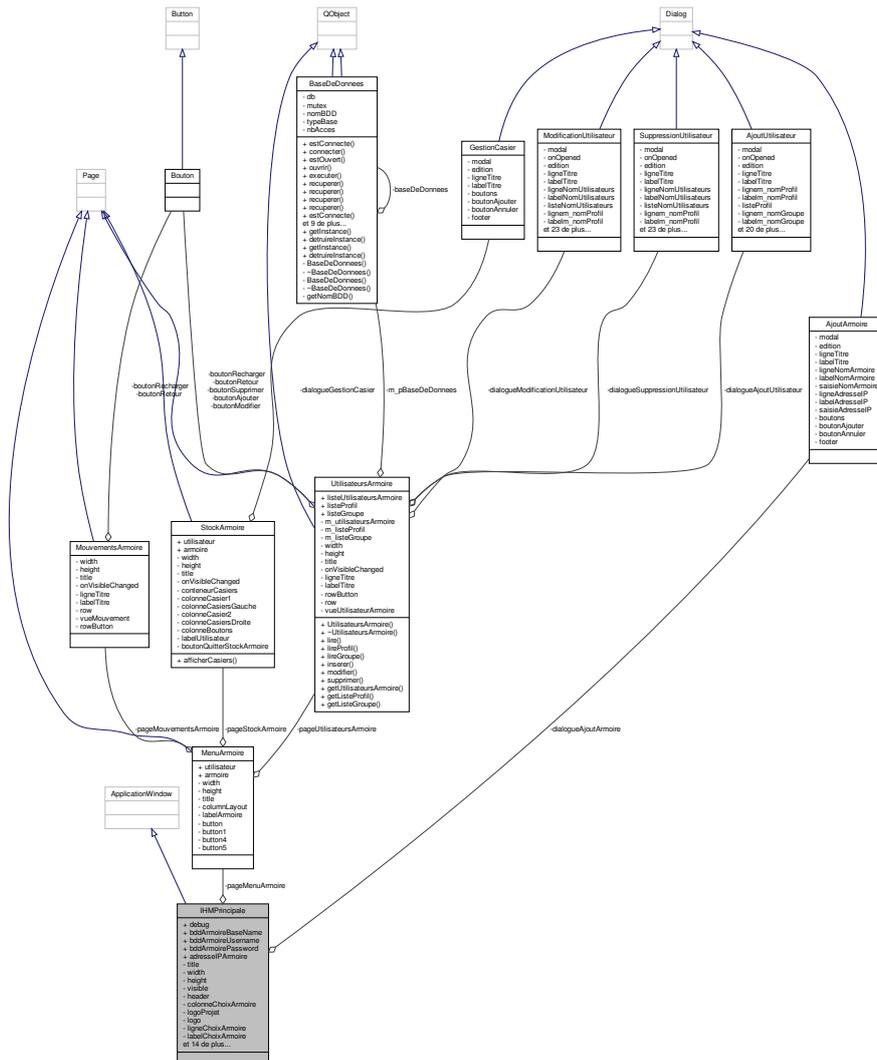
La documentation de cette classe a été générée à partir des fichiers suivants :

- [IhmGestionGroupe.h](#)
- [IhmGestionGroupe.cpp](#)

9.20 Référence de la classe IHMPPrincipale

La fenêtre principale de l'application.

Graphe de collaboration de IHMPPrincipale :



Propriétés

- bool `debug`
- string `bddArmoireBaseName`
- string `bddArmoireUsername`
- string `bddArmoirePassword`
- string `adresseIPArmoire`

Attributs privés

- var `title`
- var `width`
- var `height`
- var `visible`
- var `header`
- *Menu de la fenêtre principale.*
- Column `colonneChoixArmoire`
- *Conteneur des éléments de l'IHMPrincipale.*
- Row `logoProjet`
- Image `logo`
- Row `ligneChoixArmoire`
- *Conteneur des éléments listeChoixArmoire et labelChoixArmoire.*
- Text `labelChoixArmoire`
- ComboBox `listeChoixArmoire`
- *Liste déroulante des armoires enregistrées.*
- Button `boutonAjouterArmoire`
- *Bouton pour ajouter une armoire.*
- Button `boutonSupprimer`
- *Bouton pour supprimer l'armoire sélectionnée.*
- Row `ligneUtilisateur`
- *Conteneur des éléments labelUtilisateur et saisieUtilisateur.*
- Text `labelUtilisateur`
- TextField `saisieUtilisateur`
- *Zone de saisie du nom d'utilisateur.*
- Row `ligneMotDePasse`
- *Conteneur des éléments labelMotDePasse et saisieMotDePasse.*
- Text `labelMotDePasse`
- TextField `saisieMotDePasse`
- *Zone de saisie du mot de passe.*
- Row `ligneConnexion`
- *Conteneur des boutons boutonAjouterArmoire et boutonSelectionArmoire.*
- Button `boutonConnecterArmoire`
- *Bouton pour sélectionner une armoire.*
- Row `ligneIndicateur`
- BusyIndicator `indicateur`
- MenuArmoire `pageMenuArmoire`
- *page de menu de l'armoire sélectionnée*
- AjoutArmoire `dialogueAjoutArmoire`
- *boîte de dialogue permettant d'ajouter une armoire à la liste*
- var `footer`
- *pied de page*

9.20.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

9.20.2 Documentation des données membres

9.20.2.1 boutonAjouterArmoire

Button IHMPPrincipale::boutonAjouterArmoire [private]

9.20.2.2 boutonConnecterArmoire

Button IHMPPrincipale::boutonConnecterArmoire [private]

9.20.2.3 boutonSupprimer

Button IHMPPrincipale::boutonSupprimer [private]

9.20.2.4 colonneChoixArmoire

Column IHMPPrincipale::colonneChoixArmoire [private]

9.20.2.5 dialogueAjoutArmoire

[AjoutArmoire](#) IHMPPrincipale::dialogueAjoutArmoire [private]

9.20.2.6 footer

var IHMPPrincipale::footer [private]

9.20.2.7 header

var IHMPPrincipale::header [private]

9.20.2.8 height

var IHMPPrincipale::height [private]

9.20.2.9 indicateur

BusyIndicator IHMPPrincipale::indicateur [private]

9.20.2.10 labelChoixArmoire

Text IHMPricipale::labelChoixArmoire [private]

9.20.2.11 labelMotDePasse

Text IHMPricipale::labelMotDePasse [private]

9.20.2.12 labelUtilisateur

Text IHMPricipale::labelUtilisateur [private]

9.20.2.13 ligneChoixArmoire

Row IHMPricipale::ligneChoixArmoire [private]

9.20.2.14 ligneConnexion

Row IHMPricipale::ligneConnexion [private]

9.20.2.15 ligneIndicateur

Row IHMPricipale::ligneIndicateur [private]

9.20.2.16 ligneMotDePasse

Row IHMPricipale::ligneMotDePasse [private]

9.20.2.17 ligneUtilisateur

Row IHMPricipale::ligneUtilisateur [private]

9.20.2.18 listeChoixArmoire

ComboBox IHMPricipale::listeChoixArmoire [private]

9.20.2.19 logo

```
Image IHMPricipale::logo [private]
```

9.20.2.20 logoProjet

```
Row IHMPricipale::logoProjet [private]
```

9.20.2.21 pageMenuArmoire

```
MenuArmoire IHMPricipale::pageMenuArmoire [private]
```

9.20.2.22 saisieMotDePasse

```
TextField IHMPricipale::saisieMotDePasse [private]
```

9.20.2.23 saisieUtilisateur

```
TextField IHMPricipale::saisieUtilisateur [private]
```

9.20.2.24 title

```
var IHMPricipale::title [private]
```

9.20.2.25 visible

```
var IHMPricipale::visible [private]
```

9.20.2.26 width

```
var IHMPricipale::width [private]
```

9.20.3 Documentation des propriétés

9.20.3.1 adresseIPArmoire

string IHMPrincipale::adresseIPArmoire

9.20.3.2 bddArmoireBaseName

string IHMPrincipale::bddArmoireBaseName

9.20.3.3 bddArmoirePassword

string IHMPrincipale::bddArmoirePassword

9.20.3.4 bddArmoireUsername

string IHMPrincipale::bddArmoireUsername

9.20.3.5 debug

bool IHMPrincipale::debug

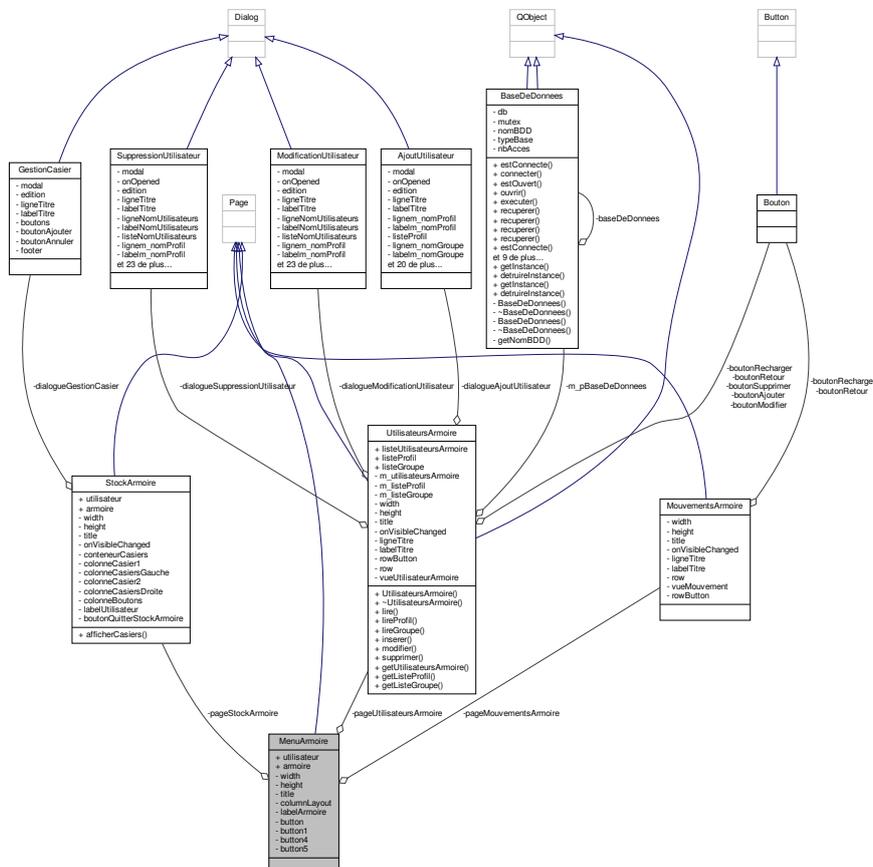
La documentation de cette classe a été générée à partir du fichier suivant :

[IHMPrincipale.qml](#)

9.21 Référence de la classe MenuArmoire

page de la gestion de l'Armoire

Graphe de collaboration de MenuArmoire :



Propriétés

- string [utilisateur](#)
- string [armoire](#)

Attributs privés

- var [width](#)
- var [height](#)
- var [title](#)
- [ColumnLayout](#) [columnLayout](#)
- [Text](#) [labelArmoire](#)
- [Button](#) [button](#)
- [Button](#) [button1](#)
- [Button](#) [button4](#)
- [Button](#) [button5](#)
- [StockArmoire](#) [pageStockArmoire](#)
- [MouvementsArmoire](#) [pageMouvementsArmoire](#)
- [UtilisateursArmoire](#) [pageUtilisateursArmoire](#)

9.21.1 Description détaillée

La page de gestion d'une armoire.

Auteur

Robin Gauthier

Version

1.1

9.21.2 Documentation des données membres

9.21.2.1 `button`

```
Button MenuArmoire::button [private]
```

9.21.2.2 `button1`

```
Button MenuArmoire::button1 [private]
```

9.21.2.3 `button4`

```
Button MenuArmoire::button4 [private]
```

9.21.2.4 button5

```
Button MenuArmoire::button5 [private]
```

9.21.2.5 columnLayout

```
ColumnLayout MenuArmoire::columnLayout [private]
```

9.21.2.6 height

```
var MenuArmoire::height [private]
```

9.21.2.7 labelArmoire

```
Text MenuArmoire::labelArmoire [private]
```

9.21.2.8 pageMouvementsArmoire

```
MouvementsArmoire MenuArmoire::pageMouvementsArmoire [private]
```

9.21.2.9 pageStockArmoire

```
StockArmoire MenuArmoire::pageStockArmoire [private]
```

9.21.2.10 pageUtilisateursArmoire

```
UtilisateursArmoire MenuArmoire::pageUtilisateursArmoire [private]
```

9.21.2.11 title

```
var MenuArmoire::title [private]
```

9.21.2.12 width

```
var MenuArmoire::width [private]
```

9.21.3 Documentation des propriétés

9.21.3.1 armoire

```
string MenuArmoire::armoire
```

9.21.3.2 utilisateur

```
string MenuArmoire::utilisateur
```

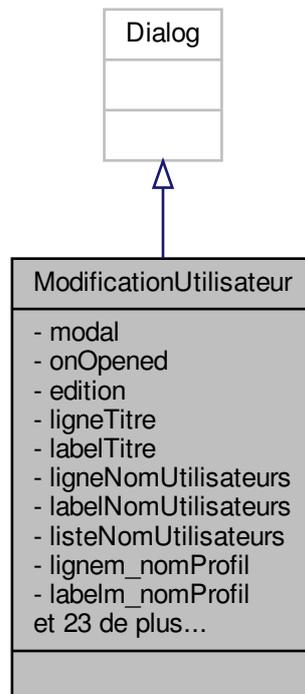
La documentation de cette classe a été générée à partir du fichier suivant :

— [MenuArmoire.qml](#)

9.22 Référence de la classe ModificationUtilisateur

La boîte de dialogue [ModificationUtilisateur](#).

Graphe de collaboration de ModificationUtilisateur :



Attributs privés

- var `modal`
- var `onOpened`
- Column `edition`
Conteneur des éléments de la page qml [AjoutArmoire](#).
- Row `ligneTitre`
Conteneur du texte `labelTitre`.
- Text `labelTitre`
titre de la boîte de dialogue
- Row `ligneNomUtilisateurs`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelNomUtilisateurs`
- ComboBox `listeNomUtilisateurs`
Liste déroulante.
- Row `lignem_nomProfil`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_nomProfil`
- ComboBox `listeProfil`
Liste déroulante du [Profil](#).
- Row `lignem_nomGroupe`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_nomGroupe`
- ComboBox `listeGroupe`
Zone de saisie du nom de l'`idGroupe`.
- Row `lignem_nom`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_nom`
- TextField `saisiem_nom`
Zone de saisie du nom de l'[armoire](#).
- Row `lignem_prenom`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_prenom`
- TextField `saisiem_prenom`
Zone de saisie du nom de l'[armoire](#).
- Row `lignem_dateDeValidite`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_dateDeValidite`
- TextField `saisiem_dateDeValidite`
Zone de saisie du nom de l'[armoire](#).
- Row `lignem_identifiant`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_identifiant`
- TextField `saisiem_identifiant`
Zone de saisie du nom de l'[armoire](#).
- Row `lignem_badge`
Conteneur des éléments `lignem_badge` et `saisiem_badge`.
- Text `labelm_badge`
- TextField `saisiem_badge`
Zone de saisie du badge.
- Row `boutons`
Conteneur des boutons `boutonAjouter` et `boutonAnnuler`.
- Button `boutonModifier`
Bouton pour modifier un utilisateur.
- Button `boutonAnnuler`
Bouton pour revenir a la page [IHMPricipale](#).
- var `footer`
 pied de page

9.22.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

9.22.2 Documentation des données membres

9.22.2.1 boutonAnnuler

Button ModificationUtilisateur::boutonAnnuler [private]

9.22.2.2 boutonModifier

Button ModificationUtilisateur::boutonModifier [private]

9.22.2.3 boutons

Row ModificationUtilisateur::boutons [private]

9.22.2.4 edition

Column ModificationUtilisateur::edition [private]

9.22.2.5 footer

var ModificationUtilisateur::footer [private]

9.22.2.6 labelm_badge

Text ModificationUtilisateur::labelm_badge [private]

9.22.2.7 labelm_dateDeValidite

Text ModificationUtilisateur::labelm_dateDeValidite [private]

9.22.2.8 labelm_identifiant

Text ModificationUtilisateur::labelm_identifiant [private]

9.22.2.9 labelm_nom

Text ModificationUtilisateur::labelm_nom [private]

9.22.2.10 labelm_nomGroupe

Text ModificationUtilisateur::labelm_nomGroupe [private]

9.22.2.11 labelm_nomProfil

Text ModificationUtilisateur::labelm_nomProfil [private]

9.22.2.12 labelm_prenom

Text ModificationUtilisateur::labelm_prenom [private]

9.22.2.13 labelNomUtilisateurs

Text ModificationUtilisateur::labelNomUtilisateurs [private]

9.22.2.14 labelTitre

Text ModificationUtilisateur::labelTitre [private]

9.22.2.15 lignem_badge

Row ModificationUtilisateur::lignem_badge [private]

9.22.2.16 lignem_dateDeValidite

Row ModificationUtilisateur::lignem_dateDeValidite [private]

9.22.2.17 lignem_identifiant

Row ModificationUtilisateur::lignem_identifiant [private]

9.22.2.18 lignem_nom

Row ModificationUtilisateur::lignem_nom [private]

9.22.2.19 lignem_nomGroupe

Row ModificationUtilisateur::lignem_nomGroupe [private]

9.22.2.20 lignem_nomProfil

Row ModificationUtilisateur::lignem_nomProfil [private]

9.22.2.21 lignem_prenom

Row ModificationUtilisateur::lignem_prenom [private]

9.22.2.22 ligneNomUtilisateurs

Row ModificationUtilisateur::ligneNomUtilisateurs [private]

9.22.2.23 ligneTitre

Row ModificationUtilisateur::ligneTitre [private]

9.22.2.24 listeGroupe

ComboBox ModificationUtilisateur::listeGroupe [private]

9.22.2.25 listeNomUtilisateurs

ComboBox ModificationUtilisateur::listeNomUtilisateurs [private]

9.22.2.26 listeProfil

ComboBox ModificationUtilisateur::listeProfil [private]

9.22.2.27 modal

```
var ModificationUtilisateur::modal [private]
```

9.22.2.28 onOpened

```
var ModificationUtilisateur::onOpened [private]
```

9.22.2.29 saisiem_badge

```
TextField ModificationUtilisateur::saisiem_badge [private]
```

9.22.2.30 saisiem_dateDeValidite

```
TextField ModificationUtilisateur::saisiem_dateDeValidite [private]
```

9.22.2.31 saisiem_identifiant

```
TextField ModificationUtilisateur::saisiem_identifiant [private]
```

9.22.2.32 saisiem_nom

```
TextField ModificationUtilisateur::saisiem_nom [private]
```

9.22.2.33 saisiem_prenom

```
TextField ModificationUtilisateur::saisiem_prenom [private]
```

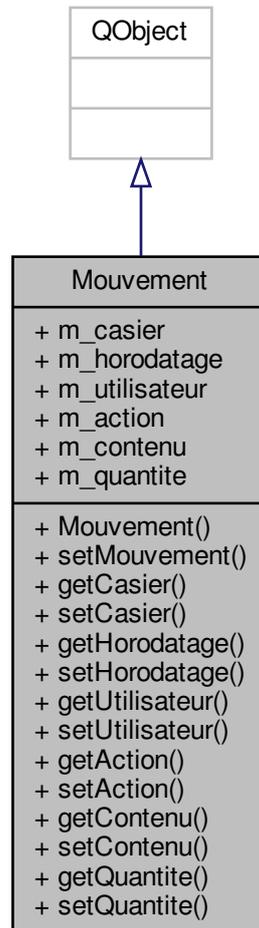
La documentation de cette classe a été générée à partir du fichier suivant :

— [ModificationUtilisateur.qml](#)

9.23 Référence de la classe Mouvement

```
#include <mouvement.h>
```

Graphe de collaboration de Mouvement :



Signaux

- void [casierChanged](#) ()
- void [horodatageChanged](#) ()
- void [utilisateurChanged](#) ()
- void [actionChanged](#) ()
- void [contenuChanged](#) ()
- void [quantiteChanged](#) ()

Fonctions membres publiques

- [Mouvement](#) (QString p_casier="", QString p_horodatage="", QString p_utilisateur="", QString p_action="", QString p_↔
contenu="", QString p_quantite="", QObject *p_parent=nullptr)
- void [setMouvement](#) (QStringList p_mouvement)
- QString [getCasier](#) () const
- void [setCasier](#) (QString p_casier)

- QString `getHorodatage` () const
- void `setHorodatage` (QString p_horodatage)
- QString `getUtilisateur` () const
- void `setUtilisateur` (QString p_utilisateur)
- QString `getAction` () const
- void `setAction` (QString p_action)
- QString `getContenu` () const
- void `setContenu` (QString p_contenu)
- QString `getQuantite` () const
- void `setQuantite` (QString quantite)

Propriétés

- QString `m_casier`
- QString `m_horodatage`
- QString `m_utilisateur`
- QString `m_action`
- QString `m_contenu`
- QString `m_quantite`

9.23.1 Documentation des constructeurs et destructeur

9.23.1.1 Mouvement()

```
Mouvement::Mouvement (
    QString p_casier = "",
    QString p_horodatage = "",
    QString p_utilisateur = "",
    QString p_action = "",
    QString p_contenu = "",
    QString p_quantite = "",
    QObject * p_parent = nullptr ) [explicit]

00005 : QObject(p_parent), m_casier(p_casier), m_horodatage(p_horodatage),
      m_utilisateur(p_utilisateur),
00006     m_action(p_action), m_contenu(p_contenu), m_quantite(p_quantite)
00007 {
00008 }
```

9.23.2 Documentation des fonctions membres

9.23.2.1 actionChanged

```
void Mouvement::actionChanged ( ) [signal]
```

9.23.2.2 casierChanged

```
void Mouvement::casierChanged ( ) [signal]
```

9.23.2.3 contenuChanged

```
void Mouvement::contenuChanged ( ) [signal]
```

9.23.2.4 getAction()

```
QString Mouvement::getAction ( ) const
```

Références [m_action](#).

```
00054 {  
00055     return m_action;  
00056 }
```

9.23.2.5 getCasier()

```
QString Mouvement::getCasier ( ) const
```

Références [m_casier](#).

```
00024 {  
00025     return m_casier;  
00026 }
```

9.23.2.6 getContenu()

```
QString Mouvement::getContenu ( ) const
```

Références [m_contenu](#).

```
00064 {  
00065     return m_contenu;  
00066 }
```

9.23.2.7 getHorodatage()

```
QString Mouvement::getHorodatage ( ) const
```

Références [m_horodatage](#).

```
00034 {  
00035     return m_horodatage;  
00036 }
```

9.23.2.8 getQuantite()

```
QString Mouvement::getQuantite ( ) const
```

Références [m_quantite](#).

```
00074 {  
00075     return m_quantite;  
00076 }
```

9.23.2.9 getUtilisateur()

```
QString Mouvement::getUtilisateur ( ) const
```

Références [m_utilisateur](#).

```
00044 {  
00045     return m_utilisateur;  
00046 }
```

9.23.2.10 horodatageChanged

```
void Mouvement::horodatageChanged ( ) [signal]
```

9.23.2.11 quantiteChanged

```
void Mouvement::quantiteChanged ( ) [signal]
```

9.23.2.12 setAction()

```
void Mouvement::setAction (  
    QString p_action )
```

Références [m_action](#).

```
00059 {  
00060     m_action = p_action;  
00061 }
```

9.23.2.13 setCasier()

```
void Mouvement::setCasier (
    QString p_casier )
```

Références [m_casier](#).

```
00029 {
00030     m_casier = p_casier;
00031 }
```

9.23.2.14 setContenu()

```
void Mouvement::setContenu (
    QString p_contenu )
```

Références [m_contenu](#).

```
00069 {
00070     m_contenu = p_contenu;
00071 }
```

9.23.2.15 setHorodatage()

```
void Mouvement::setHorodatage (
    QString p_horodatage )
```

Références [m_horodatage](#).

```
00039 {
00040     m_horodatage = p_horodatage;
00041 }
```

9.23.2.16 setMouvement()

```
void Mouvement::setMouvement (
    QStringList p_mouvement )
```

Références [m_action](#), [m_casier](#), [m_contenu](#), [m_horodatage](#), [m_quantite](#), et [m_utilisateur](#).

```
00011 {
00012     if (p_mouvement.size() == 6)
00013     {
00014         m_casier = p_mouvement.at(0);
00015         m_horodatage = p_mouvement.at(1);
00016         m_utilisateur = p_mouvement.at(2);
00017         m_action = p_mouvement.at(3);
00018         m_contenu = p_mouvement.at(4);
00019         m_quantite = p_mouvement.at(5);
00020     }
00021 }
```

9.23.2.17 setQuantite()

```
void Mouvement::setQuantite (
    QString quantite )
```

Références [m_quantite](#).

```
00079 {
00080     m_quantite = p_quantite;
00081 }
```

9.23.2.18 setUtilisateur()

```
void Mouvement::setUtilisateur (
    QString p_utilisateur )
```

Références [m_utilisateur](#).

```
00049 {
00050     m_utilisateur = p_utilisateur;
00051 }
```

9.23.2.19 utilisateurChanged

```
void Mouvement::utilisateurChanged ( ) [signal]
```

9.23.3 Documentation des propriétés

9.23.3.1 m_action

```
QString Mouvement::m_action [read], [write]
```

Référencé par [getAction\(\)](#), [setAction\(\)](#), et [setMouvement\(\)](#).

9.23.3.2 m_casier

```
QString Mouvement::m_casier [read], [write]
```

Référencé par [getCasier\(\)](#), [setCasier\(\)](#), et [setMouvement\(\)](#).

9.23.3.3 m_contenu

QString Mouvement::m_contenu [read], [write]

Référencé par [getContenu\(\)](#), [setContenu\(\)](#), et [setMouvement\(\)](#).

9.23.3.4 m_horodatage

QString Mouvement::m_horodatage [read], [write]

Référencé par [getHorodatage\(\)](#), [setHorodatage\(\)](#), et [setMouvement\(\)](#).

9.23.3.5 m_quantite

QString Mouvement::m_quantite [read], [write]

Référencé par [getQuantite\(\)](#), [setMouvement\(\)](#), et [setQuantite\(\)](#).

9.23.3.6 m_utilisateur

QString Mouvement::m_utilisateur [read], [write]

Référencé par [getUtilisateur\(\)](#), [setMouvement\(\)](#), et [setUtilisateur\(\)](#).

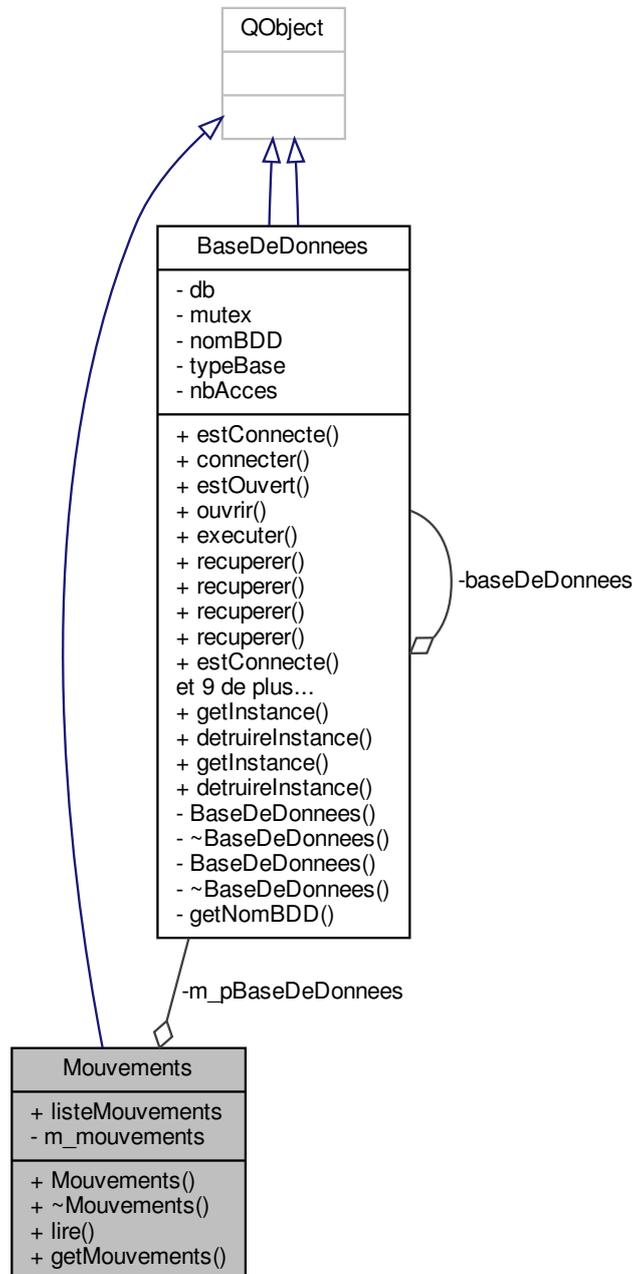
La documentation de cette classe a été générée à partir des fichiers suivants :

- [mouvement.h](#)
- [mouvement.cpp](#)

9.24 Référence de la classe Mouvements

```
#include <mouvements.h>
```

Grappe de collaboration de Mouvements :



Signaux

— void `listeMouvementsChanged` ()

Fonctions membres publiques

— `Mouvements` (QObject *p_parent=nullptr)
 — `~Mouvements` ()
 — Q_INVOKABLE bool `lire` ()
 — QVariant `getMouvements` ()

Propriétés

— QVariant [listeMouvements](#)

Attributs privés

— QList< QObject * > [m_mouvements](#)
 — [BaseDeDonnees](#) * [m_pBaseDeDonnees](#)

9.24.1 Documentation des constructeurs et destructeur

9.24.1.1 Mouvements()

```
Mouvements::Mouvements (
    QObject * p_parent = nullptr ) [explicit]
```

Références [BaseDeDonnees : :getInstance\(\)](#), et [m_pBaseDeDonnees](#).

```
00005             : QObject (p_parent)
00006 {
00007     m_pBaseDeDonnees = BaseDeDonnees::getInstance ("MYSQL");
00008 }
```

9.24.1.2 ~Mouvements()

```
Mouvements::~~Mouvements ( )
```

Références [BaseDeDonnees : :destruireInstance\(\)](#).

```
00011 {
00012     BaseDeDonnees::destruireInstance ("MYSQL");
00013 }
```

9.24.2 Documentation des fonctions membres

9.24.2.1 getMouvements()

```
QVariant Mouvements::getMouvements ( )
```

Références [m_mouvements](#).

```
00041 {
00042     return QVariant::fromValue (m_mouvements);
00043 }
```

9.24.2.2 lire()

```
bool Mouvements::lire ( )
```

Références [listeMouvementsChanged\(\)](#), [m_mouvements](#), [m_pBaseDeDonnees](#), et [BaseDeDonnees : :recuperer\(\)](#).

```
00016 {
00017     QString requete = "SELECT Stock.numeroCasier, Mouvement.Horodatage, Utilisateur.Nom, Action.Nom,
Article.Nom, Mouvement.Quantite FROM Mouvement INNER JOIN Stock ON Mouvement.idStock = Stock.idStock INNER JOIN
Utilisateur ON Mouvement.idUtilisateur = Utilisateur.idUtilisateur INNER JOIN Action ON Mouvement.idAction =
Action.idAction INNER JOIN Article ON Mouvement.idStock = Stock.idStock AND Stock.idArticle =
Article.idArticle";
00018     QVector<QStringList> donneesMouvements;
00019
00020     if (m_pBaseDeDonnees->recuperer(requete, donneesMouvements))
00021     {
00022         QStringList donneesMouvement;
00023         m_mouvements.clear();
00024
00025         for(int i = 0; i < donneesMouvements.size(); i++)
00026         {
00027             donneesMouvement = donneesMouvements.at(i);
00028             qDebug() << Q_FUNC_INFO << "donneesMouvement" << donneesMouvement;
00029             Mouvement *pMouvement = new Mouvement (donneesMouvement.at(0),
QDateTime::fromString(donneesMouvement.at(1), "yyyy-MM-ddTHH:mm:ss").toString("dd/MM/yyyy HH:mm"), donneesMouvement.at(2),
donneesMouvement.at(3), donneesMouvement.at(4), donneesMouvement.at(5));
00030             m_mouvements.append(pMouvement);
00031         }
00032         emit listeMouvementsChanged();
00033     }
00034     return true;
00035 }
00036
00037 return false;
00038 }
```

9.24.2.3 listeMouvementsChanged

```
void Mouvements::listeMouvementsChanged ( ) [signal]
```

Référencé par [lire\(\)](#).

9.24.3 Documentation des données membres

9.24.3.1 m_mouvements

```
QList<QObject*> Mouvements::m_mouvements [private]
```

Référencé par [getMouvements\(\)](#), et [lire\(\)](#).

9.24.3.2 m_pBaseDeDonnees

```
BaseDeDonnees* Mouvements::m_pBaseDeDonnees [private]
```

Référencé par [lire\(\)](#), et [Mouvements\(\)](#).

9.24.4 Documentation des propriétés

9.24.4.1 listeMouvements

```
QVariant Mouvements::listeMouvements [read]
```

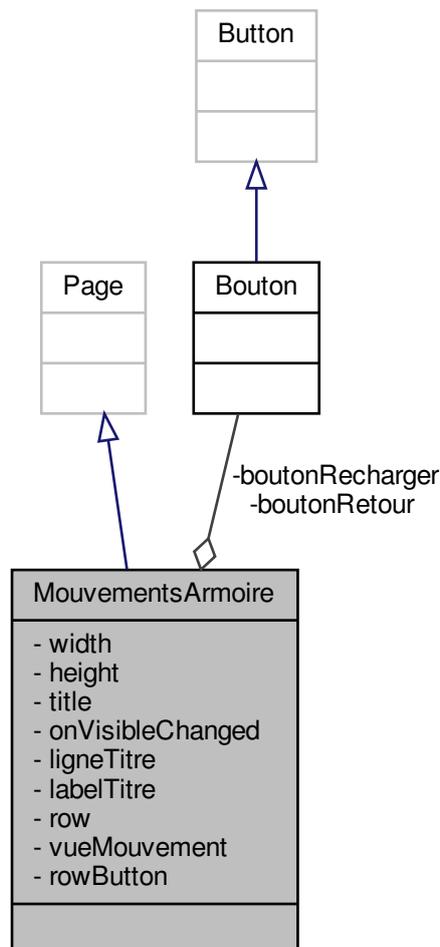
La documentation de cette classe a été générée à partir des fichiers suivants :

- [mouvements.h](#)
- [mouvements.cpp](#)

9.25 Référence de la classe MouvementsArmoire

page des mouvements d'une armoire

Graphe de collaboration de MouvementsArmoire :



Attributs privés

- var `width`
- var `height`
- var `title`
- var `onVisibleChanged`
- Row `ligneTitre`
- Text `labelTitre`
Conteneur du texte labelTitre.
- Row `row`
titre de la boîte de dialogue
- TableView `vueMouvement`
- Row `rowButton`
- Bouton `boutonRecharger`
- Bouton `boutonRetour`

9.25.1 Description détaillée

La page des mouvements d'une armoire.

Auteur

Robin Gauthier

Version

1.1

9.25.2 Documentation des données membres**9.25.2.1 boutonRecharger**

`Bouton` `MouvementsArmoire::boutonRecharger` [private]

9.25.2.2 boutonRetour

`Bouton` `MouvementsArmoire::boutonRetour` [private]

9.25.2.3 height

`var` `MouvementsArmoire::height` [private]

9.25.2.4 labelTitre

`Text` `MouvementsArmoire::labelTitre` [private]

9.25.2.5 ligneTitre

```
Row MouvementsArmoire::ligneTitre [private]
```

9.25.2.6 onVisibleChanged

```
var MouvementsArmoire::onVisibleChanged [private]
```

9.25.2.7 row

```
Row MouvementsArmoire::row [private]
```

9.25.2.8 rowButton

```
Row MouvementsArmoire::rowButton [private]
```

9.25.2.9 title

```
var MouvementsArmoire::title [private]
```

9.25.2.10 vueMouvement

```
TableView MouvementsArmoire::vueMouvement [private]
```

9.25.2.11 width

```
var MouvementsArmoire::width [private]
```

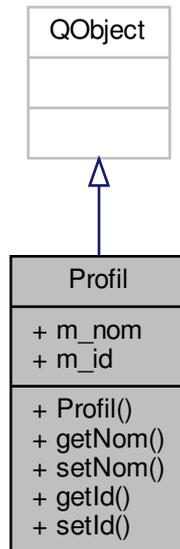
La documentation de cette classe a été générée à partir du fichier suivant :

— [MouvementsArmoire.qml](#)

9.26 Référence de la classe Profil

```
#include <profil.h>
```

Graphe de collaboration de Profil :



Signaux

- void [nomChanged](#) ()
- void [idChanged](#) ()

Fonctions membres publiques

- [Profil](#) (QString p_id="", QString p_nom="", QObject *parent=nullptr)
- QString [getNom](#) () const
- void [setNom](#) (QString p_nom)
- QString [getId](#) () const
- void [setId](#) (QString p_id)

Propriétés

- QString [m_nom](#)
- QString [m_id](#)

9.26.1 Documentation des constructeurs et destructeur

9.26.1.1 Profil()

```
Profil::Profil (
    QString p_id = "",
    QString p_nom = "",
    QObject * parent = nullptr ) [explicit]

00003                                     : QObject (parent),
    m_nom(p_nom), m_id(p_id)
00004 {
00005 }
```

9.26.2 Documentation des fonctions membres

9.26.2.1 getId()

```
QString Profil::getId ( ) const
```

Références [m_id](#).

```
00018 {
00019     return m_id;
00020 }
```

9.26.2.2 getNom()

```
QString Profil::getNom ( ) const
```

Références [m_nom](#).

```
00008 {
00009     return m_nom;
00010 }
```

9.26.2.3 idChanged

```
void Profil::idChanged ( ) [signal]
```

9.26.2.4 nomChanged

```
void Profil::nomChanged ( ) [signal]
```

9.26.2.5 setId()

```
void Profil::setId (
    QString p_id )
```

Références [m_id](#).

```
00023 {
00024     m_id = p_id;
00025 }
```

9.26.2.6 setNom()

```
void Profil::setNom (
    QString p_nom )
```

Références [m_nom](#).

```
00013 {
00014     m_nom = p_nom;
00015 }
```

9.26.3 Documentation des propriétés

9.26.3.1 m_id

QString Profil::m_id [read], [write]

Référencé par [getId\(\)](#), et [setId\(\)](#).

9.26.3.2 m_nom

QString Profil::m_nom [read], [write]

Référencé par [getNom\(\)](#), et [setNom\(\)](#).

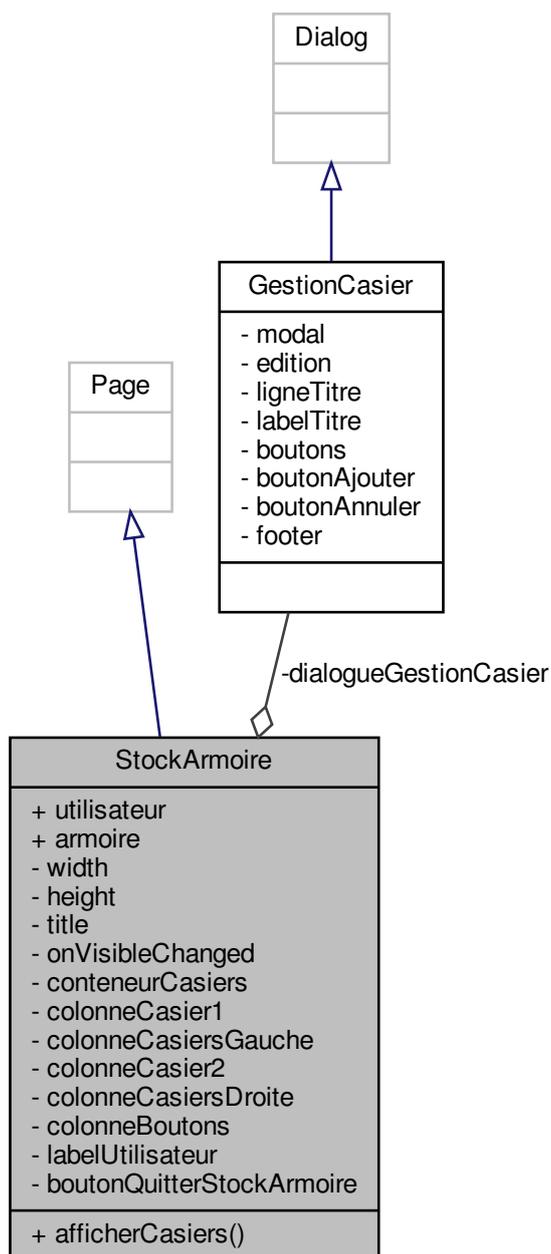
La documentation de cette classe a été générée à partir des fichiers suivants :

- [profil.h](#)
- [profil.cpp](#)

9.27 Référence de la classe StockArmoire

page de la gestion de l'[Armoire](#)

Grphe de collaboration de StockArmoire :



Fonctions membres publiques

— void [afficherCasiers](#) (nbCasiers)

Propriétés

- string `utilisateur`
- string `armoire`

Attributs privés

- var `width`
Conteneur des casiers.
- var `height`
- var `title`
- var `onVisibleChanged`
- Rectangle `conteneurCasiers`
- Column `colonneCasier1`
Colonne contenant les 4 casiers impairs.
- Repeater `colonneCasiersGauche`
- Column `colonneCasier2`
Colonne contenant les 4 derniers casiers pairs.
- Repeater `colonneCasiersDroite`
- Column `colonneBoutons`
Colonne contenant le bouton `boutonQuitterStockArmoire`.
- Text `labelUtilisateur`
- Button `boutonQuitterStockArmoire`
bouton pour quitter la page `StockArmoire` (bouton retour et non annuler pour l'instant)
- `GestionCasier dialogueGestionCasier`
boîte de dialogue permettant de gérer un casier

9.27.1 Description détaillée

La page de gestion d'une armoire.

Auteur

Robin Gauthier

Version

1.1

9.27.2 Documentation des fonctions membres

9.27.2.1 `afficherCasiers()`

```
void StockArmoire::afficherCasiers (
    nbCasiers )
```

9.27.3 Documentation des données membres

9.27.3.1 `boutonQuitterStockArmoire`

```
Button StockArmoire::boutonQuitterStockArmoire [private]
```

9.27.3.2 colonneBoutons

Column StockArmoire::colonneBoutons [private]

9.27.3.3 colonneCasier1

Column StockArmoire::colonneCasier1 [private]

9.27.3.4 colonneCasier2

Column StockArmoire::colonneCasier2 [private]

9.27.3.5 colonneCasiersDroite

Repeater StockArmoire::colonneCasiersDroite [private]

9.27.3.6 colonneCasiersGauche

Repeater StockArmoire::colonneCasiersGauche [private]

9.27.3.7 conteneurCasiers

Rectangle StockArmoire::conteneurCasiers [private]

9.27.3.8 dialogueGestionCasier

GestionCasier StockArmoire::dialogueGestionCasier [private]

9.27.3.9 height

var StockArmoire::height [private]

9.27.3.10 labelUtilisateur

Text StockArmoire::labelUtilisateur [private]

9.27.3.11 onVisibleChanged

```
var StockArmoire::onVisibleChanged [private]
```

9.27.3.12 title

```
var StockArmoire::title [private]
```

9.27.3.13 width

```
var StockArmoire::width [private]
```

9.27.4 Documentation des propriétés

9.27.4.1 armoire

```
string StockArmoire::armoire
```

9.27.4.2 utilisateur

```
string StockArmoire::utilisateur
```

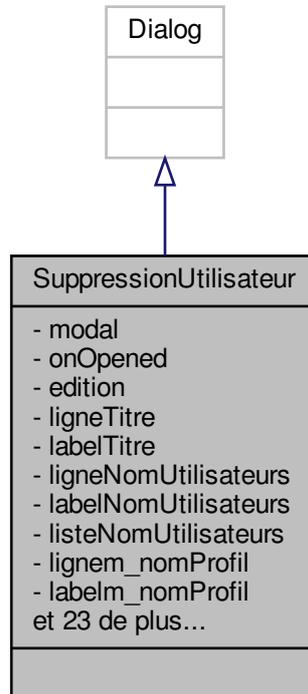
La documentation de cette classe a été générée à partir du fichier suivant :

— [StockArmoire.qml](#)

9.28 Référence de la classe SuppressionUtilisateur

La boîte de dialogue [SuppressionUtilisateur](#).

Graphe de collaboration de SuppressionUtilisateur :



Attributs privés

- var `modal`
- var `onOpened`
- Column `edition`
Conteneur des éléments de la page qml [AjoutArmoire](#).
- Row `ligneTitre`
Conteneur du texte `labelTitre`.
- Text `labelTitre`
titre de la boîte de dialogue
- Row `ligneNomUtilisateurs`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelNomUtilisateurs`
- ComboBox `listeNomUtilisateurs`
Liste déroulante des nom des utilisateurs.
- Row `lignem_nomProfil`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_nomProfil`
- ComboBox `listeProfil`
Liste déroulante du `Profil`.
- Row `lignem_nomGroupe`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_nomGroupe`
- ComboBox `listeGroupe`
Zone de saisie du nom de l'idGroupe.
- Row `lignem_nom`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.
- Text `labelm_nom`
- TextField `saisiem_nom`
Zone de saisie du nom de l'armoire.
- Row `lignem_prenom`
Conteneur des éléments `ligneidUtilisateur` et `saisieidUtilisateur`.

- Text [labelm_prenom](#)
- TextField [saisiem_prenom](#)
Zone de saisie du nom de l'armoire.
- Row [lignem_dateDeValidite](#)
Conteneur des éléments `lignem_dateDeValidite` et `saisiem_dateDeValidite`.
- Text [labelm_dateDeValidite](#)
- TextField [saisiem_dateDeValidite](#)
Zone de saisie du nom de l'armoire.
- Row [lignem_identifiant](#)
Conteneur des éléments `lignem_identifiant` et `saisiem_identifiant`.
- Text [labelm_identifiant](#)
- TextField [saisiem_identifiant](#)
Zone de saisie du nom de l'armoire.
- Row [lignem_badge](#)
Conteneur des éléments `lignem_badge` et `saisiem_badge`.
- Text [labelm_badge](#)
- TextField [saisiem_badge](#)
Zone de saisie du badge.
- Row [boutons](#)
Conteneur des boutons `boutonAjouter` et `boutonAnnuler`.
- Button [boutonSupprimer](#)
Bouton pour modifier un utilisateur.
- Button [boutonAnnuler](#)
Bouton pour revenir à la page [IHMPricipale](#).
- var [footer](#)
 pied de page

9.28.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

9.28.2 Documentation des données membres

9.28.2.1 boutonAnnuler

Button `SuppressionUtilisateur::boutonAnnuler` [private]

9.28.2.2 boutons

Row `SuppressionUtilisateur::boutons` [private]

9.28.2.3 boutonSupprimer

Button `SuppressionUtilisateur::boutonSupprimer` [private]

9.28.2.4 `edition`

Column `SuppressionUtilisateur::edition` [private]

9.28.2.5 `footer`

var `SuppressionUtilisateur::footer` [private]

9.28.2.6 `labelm_badge`

Text `SuppressionUtilisateur::labelm_badge` [private]

9.28.2.7 `labelm_dateDeValidite`

Text `SuppressionUtilisateur::labelm_dateDeValidite` [private]

9.28.2.8 `labelm_identifiant`

Text `SuppressionUtilisateur::labelm_identifiant` [private]

9.28.2.9 `labelm_nom`

Text `SuppressionUtilisateur::labelm_nom` [private]

9.28.2.10 `labelm_nomGroupe`

Text `SuppressionUtilisateur::labelm_nomGroupe` [private]

9.28.2.11 `labelm_nomProfil`

Text `SuppressionUtilisateur::labelm_nomProfil` [private]

9.28.2.12 `labelm_prenom`

Text `SuppressionUtilisateur::labelm_prenom` [private]

9.28.2.13 labelNomUtilisateurs

Text SuppressionUtilisateur::labelNomUtilisateurs [private]

9.28.2.14 labelTitre

Text SuppressionUtilisateur::labelTitre [private]

9.28.2.15 lignem_badge

Row SuppressionUtilisateur::lignem_badge [private]

9.28.2.16 lignem_dateDeValidite

Row SuppressionUtilisateur::lignem_dateDeValidite [private]

9.28.2.17 lignem_identifiant

Row SuppressionUtilisateur::lignem_identifiant [private]

9.28.2.18 lignem_nom

Row SuppressionUtilisateur::lignem_nom [private]

9.28.2.19 lignem_nomGroupe

Row SuppressionUtilisateur::lignem_nomGroupe [private]

9.28.2.20 lignem_nomProfil

Row SuppressionUtilisateur::lignem_nomProfil [private]

9.28.2.21 lignem_prenom

Row SuppressionUtilisateur::lignem_prenom [private]

9.28.2.22 `ligneNomUtilisateurs`

```
Row SuppressionUtilisateur::ligneNomUtilisateurs [private]
```

9.28.2.23 `ligneTitre`

```
Row SuppressionUtilisateur::ligneTitre [private]
```

9.28.2.24 `listeGroupe`

```
ComboBox SuppressionUtilisateur::listeGroupe [private]
```

9.28.2.25 `listeNomUtilisateurs`

```
ComboBox SuppressionUtilisateur::listeNomUtilisateurs [private]
```

9.28.2.26 `listeProfil`

```
ComboBox SuppressionUtilisateur::listeProfil [private]
```

9.28.2.27 `modal`

```
var SuppressionUtilisateur::modal [private]
```

9.28.2.28 `onOpened`

```
var SuppressionUtilisateur::onOpened [private]
```

9.28.2.29 `saisiem_badge`

```
TextField SuppressionUtilisateur::saisiem_badge [private]
```

9.28.2.30 `saisiem_dateDeValidite`

```
TextField SuppressionUtilisateur::saisiem_dateDeValidite [private]
```

9.28.2.31 saisie_identifiant

```
TextField SuppressionUtilisateur::saisie_identifiant [private]
```

9.28.2.32 saisie_nom

```
TextField SuppressionUtilisateur::saisie_nom [private]
```

9.28.2.33 saisie_prenom

```
TextField SuppressionUtilisateur::saisie_prenom [private]
```

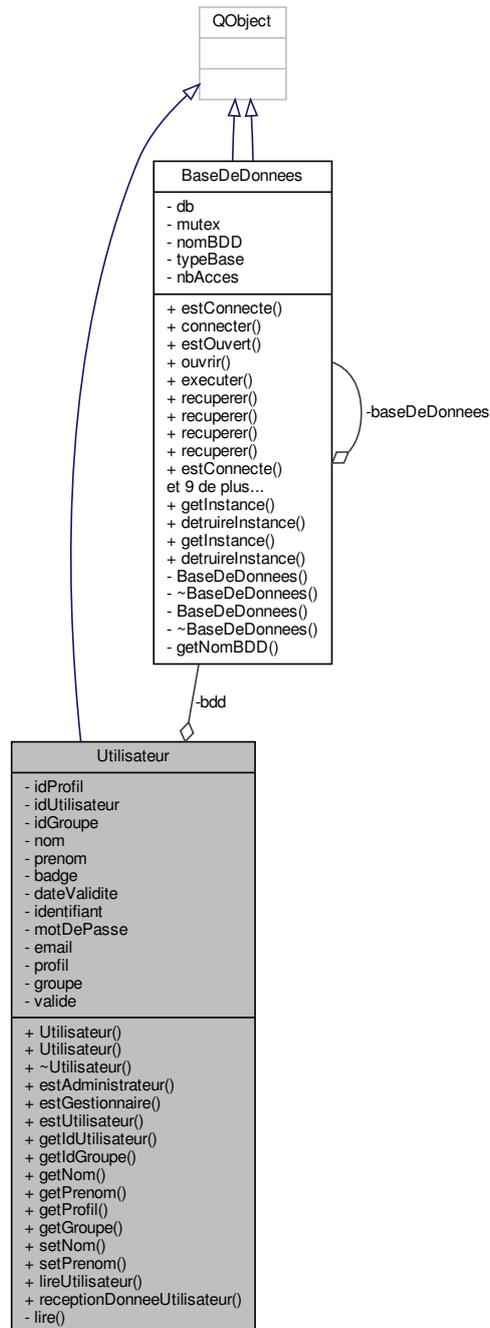
La documentation de cette classe a été générée à partir du fichier suivant :

— [SuppressionUtilisateur.qml](#)

9.29 Référence de la classe Utilisateur

```
#include <utilisateur.h>
```

Grphe de collaboration de Utilisateur :



Connecteurs publics

— void `receptionDonneeUtilisateur` (QString `nom`, QString `motDePasse`)

Fonctions membres publiques

— `Utilisateur` ()
 — `Utilisateur` (QString `idUtilisateur`)
 — `~Utilisateur` ()

- bool `estAdministrateur` () const
- bool `estGestionnaire` () const
- bool `estUtilisateur` () const
- QString `getIdUtilisateur` () const
- QString `getIdGroupe` () const
- QString `getNom` () const
- QString `getPrenom` () const
- QString `getProfil` () const
- QString `getGroupe` () const
- void `setNom` (QString `nom`)
- void `setPrenom` (QString `prenom`)
- bool `lireUtilisateur` (QString `identifiant`, QString `password`)

Fonctions membres privées

- void `lire` ()

Attributs privés

- QString `idProfil`
- QString `idUtilisateur`
- QString `idGroupe`
- QString `nom`
- QString `prenom`
- QString `badge`
- QString `dateValidite`
- QString `identifiant`
- QString `motDePasse`
- QString `email`
- QString `profil`
- QString `groupe`
- bool `valide`
- `BaseDeDonnees * bdd`
agrégation de l'objet `BaseDeDonnees`

9.29.1 Documentation des constructeurs et destructeur

9.29.1.1 Utilisateur() [1/2]

`Utilisateur::Utilisateur ()`

Références `bdd`, `BaseDeDonnees : :connecter()`, `BaseDeDonnees : :estConnecte()`, et `BaseDeDonnees : :getInstance()`.

```

00004         : idUtilisateur("", nom(""), prenom(""),
            identifiant(""), motDePasse(""), email(""), valide(false)
00005 {
00006     qDebug() << Q_FUNC_INFO;
00007     bdd = BaseDeDonnees::getInstance();
00008     if(!bdd->estConnecte())
00009         bdd->connecter();
00010 }
```

9.29.1.2 Utilisateur() [2/2]

```
Utilisateur::Utilisateur (
    QString idUtilisateur )
```

Références [bdd](#), [BaseDeDonnees : :connecter\(\)](#), [BaseDeDonnees : :estConnecte\(\)](#), [BaseDeDonnees : :getInstance\(\)](#), et [lire\(\)](#).

```
00012                                     : idUtilisateur(
    idUtilisateur), nom(""), prenom(""), identifiant(""),
    motDePasse(""), email(""), valide(false)
00013 {
00014     qDebug() << Q_FUNC_INFO;
00015     bdd = BaseDeDonnees::getInstance();
00016     if(!bdd->estConnecte())
00017         bdd->connecter();
00018     lire();
00019 }
```

9.29.1.3 ~Utilisateur()

```
Utilisateur::~~Utilisateur ( )
```

Références [BaseDeDonnees : :destruireInstance\(\)](#).

```
00022 {
00023     BaseDeDonnees::destruireInstance();
00024     qDebug() << Q_FUNC_INFO;
00025 }
```

9.29.2 Documentation des fonctions membres

9.29.2.1 estAdministrateur()

```
bool Utilisateur::estAdministrateur ( ) const
```

Références [profil](#).

Référencé par [IhmGestionGroupe : :ajouterGroupe\(\)](#), [IhmGestionArmoire : :gererGroupe\(\)](#), [IhmGestionGroupe : :listerGroupe\(\)](#), [IhmGestionGroupe : :modifierGroupe\(\)](#), [IhmGestionArmoire : :setUtilisateur\(\)](#), et [IhmGestionGroupe : :supprimerGroupe\(\)](#).

```
00028 {
00029     if(profil == "Administrateur")
00030         return true;
00031     return false;
00032 }
```

9.29.2.2 estGestionnaire()

```
bool Utilisateur::estGestionnaire ( ) const
```

Références [profil](#).

Référencé par [IhmGestionGroupe : :ajouterGroupe\(\)](#), [IhmGestionArmoire : :gererGroupe\(\)](#), [IhmGestionGroupe : :listerGroupe\(\)](#), [IhmGestionGroupe : :modifierGroupe\(\)](#), [IhmGestionArmoire : :setUtilisateur\(\)](#), et [IhmGestionGroupe : :supprimerGroupe\(\)](#).

```
00035 {
00036     if(profil == "Gestionnaire")
00037         return true;
00038     return false;
00039 }
```

9.29.2.3 estUtilisateur()

```
bool Utilisateur::estUtilisateur ( ) const
```

Références [profil](#).

```
00042 {
00043     if(profil == "Utilisateur")
00044         return true;
00045     return false;
00046 }
```

9.29.2.4 getGroupe()

```
QString Utilisateur::getGroupe ( ) const
```

Références [groupe](#).

Référencé par [IhmGestionArmoire : :showEvent\(\)](#).

```
00074 {
00075     return groupe;
00076 }
```

9.29.2.5 getIdGroupe()

```
QString Utilisateur::getIdGroupe ( ) const
```

Références [idGroupe](#).

```
00054 {
00055     return idGroupe;
00056 }
```

9.29.2.6 getIdUtilisateur()

```
QString Utilisateur::getIdUtilisateur ( ) const
```

Références [idUtilisateur](#).

```
00049 {  
00050     return idUtilisateur;  
00051 }
```

9.29.2.7 getNom()

```
QString Utilisateur::getNom ( ) const
```

Références [nom](#).

Référencé par [IhmGestionGroupe : :ajouterGroupe\(\)](#), [IhmGestionArmoire : :gererGroupe\(\)](#), [IhmGestionGroupe : :listerGroupe\(\)](#), [IhmGestionGroupe : :modifierGroupe\(\)](#), [IhmGestionArmoire : :showEvent\(\)](#), et [IhmGestionGroupe : :supprimerGroupe\(\)](#).

```
00059 {  
00060     return nom;  
00061 }
```

9.29.2.8 getPrenom()

```
QString Utilisateur::getPrenom ( ) const
```

Références [prenom](#).

Référencé par [IhmGestionArmoire : :showEvent\(\)](#).

```
00064 {  
00065     return prenom;  
00066 }
```

9.29.2.9 getProfil()

```
QString Utilisateur::getProfil ( ) const
```

Références [profil](#).

```
00069 {  
00070     return profil;  
00071 }
```

9.29.2.10 lire()

```
void Utilisateur::lire ( ) [private]
```

A faire Vérifier la validité de l'utilisateur

Références [badge](#), [bdd](#), [dateValidite](#), [email](#), [groupe](#), [identifiant](#), [idGroupe](#), [idProfil](#), [idUtilisateur](#), [nom](#), [prenom](#), [profil](#), et [BaseDeDonnees : :recuperer\(\)](#).

Référencé par [Utilisateur\(\)](#).

```
00089 {
00090     QString requete = "SELECT Utilisateur.idUtilisateur, Utilisateur.Nom, Utilisateur.Prenom,
Utilisateur.DateValidite, Utilisateur.Identifiant, Utilisateur.Badge, Utilisateur.Email, Profil.idProfil, Profil.Nom AS
Profil, Groupe.idGroupe, Groupe.Nom AS Groupe FROM Utilisateur INNER JOIN Groupe ON
Groupe.idGroupe=Utilisateur.idGroupe INNER JOIN Profil ON Profil.idProfil=Utilisateur.idProfil WHERE Utilisateur.idUtilisateur=
idUtilisateur + "'";
00091     QStringList donneesUtilisateur;
00092     /*
00093     * Exemple de résultat :
00094
00095     +-----+-----+-----+-----+-----+-----+-----+-----+
00096     | idUtilisateur | Nom      | Prenom | DateValidite | Identifiant | Badge | Email      |
00097     | idProfil | Profil      | idGroupe | Groupe      |
00098     +-----+-----+-----+-----+-----+-----+-----+-----+
00099     |          7 | GAUTHIER | Robin | 2019-07-01 | gauthier.r | 4444 | rgauthier2510@gmail.com |
00100     | 3 | Utilisateur |      5 | T-BTS-SN |
00101     +-----+-----+-----+-----+-----+-----+-----+-----+
00102     */
00103     bool retour = bdd->recuperer(requete, donneesUtilisateur);
00104     if(retour)
00105     {
00106         qDebug() << Q_FUNC_INFO << "donneesUtilisateur" << donneesUtilisateur;
00107         this->idProfil = donneesUtilisateur.at(7);
00108         this->identifiant = donneesUtilisateur.at(4);
00109         this->nom = donneesUtilisateur.at(1);
00110         this->prenom = donneesUtilisateur.at(2);
00111         this->dateValidite = donneesUtilisateur.at(3);
00112         this->badge = donneesUtilisateur.at(5);
00113         this->email = donneesUtilisateur.at(6);
00114         this->profil = donneesUtilisateur.at(8);
00115         this->idGroupe = donneesUtilisateur.at(9);
00116         this->groupe = donneesUtilisateur.at(10);
00117     }
00118 }
00119 }
00120 }
```

9.29.2.11 lireUtilisateur()

```
bool Utilisateur::lireUtilisateur (
    QString identifiant,
    QString password )
```

A faire Vérifier la validité de l'utilisateur

Références [badge](#), [bdd](#), [dateValidite](#), [email](#), [groupe](#), [idGroupe](#), [idProfil](#), [idUtilisateur](#), [nom](#), [prenom](#), [profil](#), et [BaseDeDonnees : :recuperer\(\)](#).

```

00123 {
00124     QString requete = "SELECT Utilisateur.idUtilisateur, Utilisateur.Nom, Utilisateur.Prenom,
Utilisateur.DateValidite, Utilisateur.Identifiant, Utilisateur.Badge, Utilisateur.Email, Profil.idProfil, Profil.Nom AS
Profil, Groupe.idGroupe, Groupe.Nom AS Groupe FROM Utilisateur INNER JOIN Groupe ON
Groupe.idGroupe=Utilisateur.idGroupe INNER JOIN Profil ON Profil.idProfil=Utilisateur.idProfil WHERE Utilisateur.Identifiant='
identifiant + "' AND Utilisateur.MotDePasse=' " + password + "'";
00125     QStringList donneesUtilisateur;
00126     /*
00127     * Exemple de résultat :
00128
+-----+-----+-----+-----+-----+-----+-----+-----+
00129 | idUtilisateur | Nom      | Prenom | DateValidite | Identifiant | Badge | Email
idProfil | Profil      | idGroupe | Groupe      |
00130
+-----+-----+-----+-----+-----+-----+-----+-----+
00131 |          7 | GAUTHIER | Robin  | 2019-07-01   | gauthier.r | 4444  | rgauthier2510@gmail.com |
3 | Utilisateur |          5 | T-BTS-SN |
00132
+-----+-----+-----+-----+-----+-----+-----+-----+
00133     */
00134
00135     bool retour = bdd->recuperer(requete, donneesUtilisateur);
00136     if(retour)
00137     {
00138         qDebug() << Q_FUNC_INFO << "donneesUtilisateur" << donneesUtilisateur;
00139
00140         this->idUtilisateur = donneesUtilisateur.at(0);
00141         this->idProfil = donneesUtilisateur.at(7);
00142         this->identifiant = donneesUtilisateur.at(4);
00143         this->nom = donneesUtilisateur.at(1);
00144         this->prenom = donneesUtilisateur.at(2);
00145         this->dateValidite = donneesUtilisateur.at(3);
00146         this->badge = donneesUtilisateur.at(5);
00147         this->email = donneesUtilisateur.at(6);
00148         this->profil = donneesUtilisateur.at(8);
00149         this->idGroupe = donneesUtilisateur.at(9);
00150         this->groupe = donneesUtilisateur.at(10);
00151     }
00152     return retour;
00153 }
00154
00155
00156 }

```

9.29.2.12 receptionDonneeUtilisateur

```

void Utilisateur::receptionDonneeUtilisateur (
    QString nom,
    QString motDePasse ) [slot]

```

Références [motDePasse](#), et [nom](#).

```

00159 {
00160     this->nom = nom;
00161     this->motDePasse = motDePasse;
00162 }

```

9.29.2.13 setNom()

```

void Utilisateur::setNom (
    QString nom )

```

Références [nom](#).

```

00079 {
00080     this->nom = nom;
00081 }

```

9.29.2.14 setPrenom()

```
void Utilisateur::setPrenom (
    QString prenom )
```

Références [prenom](#).

```
00084 {
00085     this->prenom = prenom;
00086 }
```

9.29.3 Documentation des données membres

9.29.3.1 badge

```
QString Utilisateur::badge [private]
```

Référencé par [lire\(\)](#), et [lireUtilisateur\(\)](#).

9.29.3.2 bdd

```
BaseDeDonnees* Utilisateur::bdd [private]
```

Référencé par [lire\(\)](#), [lireUtilisateur\(\)](#), et [Utilisateur\(\)](#).

9.29.3.3 dateValidite

```
QString Utilisateur::dateValidite [private]
```

Référencé par [lire\(\)](#), et [lireUtilisateur\(\)](#).

9.29.3.4 email

```
QString Utilisateur::email [private]
```

Référencé par [lire\(\)](#), et [lireUtilisateur\(\)](#).

9.29.3.5 groupe

```
QString Utilisateur::groupe [private]
```

Référencé par [getGroupe\(\)](#), [lire\(\)](#), et [lireUtilisateur\(\)](#).

9.29.3.6 identifiant

```
QString Utilisateur::identifiant [private]
```

Référencé par [lire\(\)](#).

9.29.3.7 idGroupe

```
QString Utilisateur::idGroupe [private]
```

Référencé par [getIdGroupe\(\)](#), [lire\(\)](#), et [lireUtilisateur\(\)](#).

9.29.3.8 idProfil

```
QString Utilisateur::idProfil [private]
```

Référencé par [lire\(\)](#), et [lireUtilisateur\(\)](#).

9.29.3.9 idUtilisateur

```
QString Utilisateur::idUtilisateur [private]
```

Référencé par [getIdUtilisateur\(\)](#), [lire\(\)](#), et [lireUtilisateur\(\)](#).

9.29.3.10 motDePasse

```
QString Utilisateur::motDePasse [private]
```

Référencé par [receptionDonneeUtilisateur\(\)](#).

9.29.3.11 nom

```
QString Utilisateur::nom [private]
```

Référencé par [getNom\(\)](#), [lire\(\)](#), [lireUtilisateur\(\)](#), [receptionDonneeUtilisateur\(\)](#), et [setNom\(\)](#).

9.29.3.12 prenom

```
QString Utilisateur::prenom [private]
```

Référencé par [getPrenom\(\)](#), [lire\(\)](#), [lireUtilisateur\(\)](#), et [setPrenom\(\)](#).

9.29.3.13 profil

```
QString Utilisateur::profil [private]
```

Référencé par [estAdministrateur\(\)](#), [estGestionnaire\(\)](#), [estUtilisateur\(\)](#), [getProfil\(\)](#), [lire\(\)](#), et [lireUtilisateur\(\)](#).

9.29.3.14 valide

```
bool Utilisateur::valide [private]
```

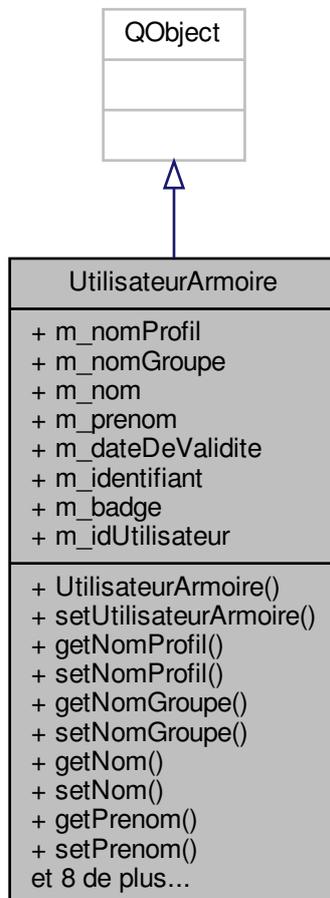
La documentation de cette classe a été générée à partir des fichiers suivants :

- [utilisateur.h](#)
- [utilisateur.cpp](#)

9.30 Référence de la classe UtilisateurArmoire

```
#include <utilisateurArmoire.h>
```

Graphique de collaboration de UtilisateurArmoire :



Signaux

- void `nomProfilChanged` ()
- void `nomGroupeChanged` ()
- void `nomChanged` ()
- void `prenomChanged` ()
- void `dateDeValiditeChanged` ()
- void `identifiantChanged` ()
- void `badgeChanged` ()
- void `idUtilisateurChanged` ()

Fonctions membres publiques

- `UtilisateurArmoire` (QString p_nomProfil="", QString p_idGroupe="", QString p_nom="", QString p_prenom="", QString p_dateDeValidite="", QString p_identifiant="", QString p_badge="", QString p_idUtilisateur="", QObject *p_parent=nullptr)
- void `setUtilisateurArmoire` (QStringList p_utilisateurArmoire)
- QString `getNomProfil` () const
- void `setNomProfil` (QString p_nomProfil)
- QString `getNomGroupe` () const
- void `setNomGroupe` (QString p_nomGroupe)
- QString `getNom` () const
- void `setNom` (QString p_nom)
- QString `getPrenom` () const
- void `setPrenom` (QString p_prenom)
- QString `getDateDeValidite` () const
- void `setDateDeValidite` (QString p_dateDeValidite)
- QString `getIdentifiant` () const
- void `setIdentifiant` (QString p_identifiant)
- QString `getBadge` () const
- void `setBadge` (QString p_badge)
- QString `getIdUtilisateur` () const
- void `setIdUtilisateur` (QString p_idUtilisateur)

Propriétés

- QString `m_nomProfil`
- QString `m_nomGroupe`
- QString `m_nom`
- QString `m_prenom`
- QString `m_dateDeValidite`
- QString `m_identifiant`
- QString `m_badge`
- QString `m_idUtilisateur`

9.30.1 Documentation des constructeurs et destructeur

9.30.1.1 UtilisateurArmoire()

```

UtilisateurArmoire::UtilisateurArmoire (
    QString p_nomProfil = "",
    QString p_idGroupe = "",
    QString p_nom = "",
    QString p_prenom = "",
    QString p_dateDeValidite = "",
    QString p_identifiant = "",
    QString p_badge = "",
    QString p_idUtilisateur = "",
    QObject * p_parent = nullptr ) [explicit]

00006     : QObject(p_parent), m_nomProfil(p_nomProfil), m_nomGroupe(p_nomGroupe),
    m_nom(p_nom), m_prenom(p_prenom),
00007     m_dateDeValidite(p_dateDeValidite), m_identifiant(p_identifiant),
    m_badge(p_badge), m_idUtilisateur(p_idUtilisateur)
00008 {
00009 }

```

9.30.2 Documentation des fonctions membres

9.30.2.1 badgeChanged

```
void UtilisateurArmoire::badgeChanged ( ) [signal]
```

9.30.2.2 dateDeValiditeChanged

```
void UtilisateurArmoire::dateDeValiditeChanged ( ) [signal]
```

9.30.2.3 getBadge()

```
QString UtilisateurArmoire::getBadge ( ) const
```

Références [m_badge](#).

```
00087 {  
00088     return m_badge;  
00089 }
```

9.30.2.4 getDateDeValidite()

```
QString UtilisateurArmoire::getDateDeValidite ( ) const
```

Références [m_dateDeValidite](#).

```
00067 {  
00068     return m_dateDeValidite;  
00069 }
```

9.30.2.5 getIdentifiant()

```
QString UtilisateurArmoire::getIdentifiant ( ) const
```

Références [m_identifiant](#).

```
00077 {  
00078     return m_identifiant;  
00079 }
```

9.30.2.6 getIdUtilisateur()

```
QString UtilisateurArmoire::getIdUtilisateur ( ) const
```

Références [m_idUtilisateur](#).

```
00097 {  
00098     return m_idUtilisateur;  
00099 }
```

9.30.2.7 getNom()

```
QString UtilisateurArmoire::getNom ( ) const
```

Références [m_nom](#).

```
00047 {  
00048     return m_nom;  
00049 }
```

9.30.2.8 getNomGroupe()

```
QString UtilisateurArmoire::getNomGroupe ( ) const
```

Références [m_nomGroupe](#).

```
00037 {  
00038     return m_nomGroupe;  
00039 }
```

9.30.2.9 getNomProfil()

```
QString UtilisateurArmoire::getNomProfil ( ) const
```

Références [m_nomProfil](#).

```
00027 {  
00028     return m_nomProfil;  
00029 }
```

9.30.2.10 getPrenom()

```
QString UtilisateurArmoire::getPrenom ( ) const
```

Références [m_prenom](#).

```
00057 {  
00058     return m_prenom;  
00059 }
```

9.30.2.11 identifiantChanged

```
void UtilisateurArmoire::identifiantChanged ( ) [signal]
```

9.30.2.12 idUtilisateurChanged

```
void UtilisateurArmoire::idUtilisateurChanged ( ) [signal]
```

9.30.2.13 nomChanged

```
void UtilisateurArmoire::nomChanged ( ) [signal]
```

9.30.2.14 nomGroupeChanged

```
void UtilisateurArmoire::nomGroupeChanged ( ) [signal]
```

9.30.2.15 nomProfilChanged

```
void UtilisateurArmoire::nomProfilChanged ( ) [signal]
```

9.30.2.16 prenomChanged

```
void UtilisateurArmoire::prenomChanged ( ) [signal]
```

9.30.2.17 setBadge()

```
void UtilisateurArmoire::setBadge (
    QString p_badge )
```

Références [m_badge](#).

```
00092 {
00093     m_badge = p_badge;
00094 }
```

9.30.2.18 setDateDeValidite()

```
void UtilisateurArmoire::setDateDeValidite (
    QString p_dateDeValidite )
```

Références [m_dateDeValidite](#).

```
00072 {
00073     m_dateDeValidite = p_dateDeValidite;
00074 }
```

9.30.2.19 setIdentifiant()

```
void UtilisateurArmoire::setIdentifiant (
    QString p_identifiant )
```

Références [m_identifiant](#).

```
00082 {
00083     m_identifiant = p_identifiant;
00084 }
```

9.30.2.20 setIdUtilisateur()

```
void UtilisateurArmoire::setIdUtilisateur (
    QString p_idUtilisateur )
```

Références [m_idUtilisateur](#).

```
00102 {
00103     m_idUtilisateur = p_idUtilisateur;
00104 }
```

9.30.2.21 setNom()

```
void UtilisateurArmoire::setNom (
    QString p_nom )
```

Références [m_nom](#).

```
00052 {
00053     m_nom = p_nom;
00054 }
```

9.30.2.22 setNomGroupe()

```
void UtilisateurArmoire::setNomGroupe (
    QString p_nomGroupe )
```

Références [m_nomGroupe](#).

```
00042 {
00043     m_nomGroupe = p_nomGroupe;
00044 }
```

9.30.2.23 setNomProfil()

```
void UtilisateurArmoire::setNomProfil (
    QString p_nomProfil )
```

Références [m_nomProfil](#).

```
00032 {
00033     m_nomProfil = p_nomProfil;
00034 }
```

9.30.2.24 setPrenom()

```
void UtilisateurArmoire::setPrenom (
    QString p_prenom )
```

Références [m_prenom](#).

```
00062 {
00063     m_prenom = p_prenom;
00064 }
```

9.30.2.25 setUtilisateurArmoire()

```
void UtilisateurArmoire::setUtilisateurArmoire (
    QStringList p_utilisateurArmoire )
```

Références [m_badge](#), [m_dateDeValidite](#), [m_identifiant](#), [m_idUtilisateur](#), [m_nom](#), [m_nomGroupe](#), [m_nomProfil](#), et [m_prenom](#).

```
00012 {
00013     if (p_utilisateurArmoire.size() == 8)
00014     {
00015         m_nomProfil = p_utilisateurArmoire.at(0);
00016         m_nomGroupe = p_utilisateurArmoire.at(1);
00017         m_nom = p_utilisateurArmoire.at(2);
00018         m_prenom = p_utilisateurArmoire.at(3);
00019         m_dateDeValidite = p_utilisateurArmoire.at(4);
00020         m_identifiant = p_utilisateurArmoire.at(5);
00021         m_badge = p_utilisateurArmoire.at(6);
00022         m_idUtilisateur = p_utilisateurArmoire.at(7);
00023     }
00024 }
```

9.30.3 Documentation des propriétés

9.30.3.1 m_badge

QString UtilisateurArmoire::m_badge [read], [write]

Référencé par [getBadge\(\)](#), [setBadge\(\)](#), et [setUtilisateurArmoire\(\)](#).

9.30.3.2 m_dateDeValidite

QString UtilisateurArmoire::m_dateDeValidite [read], [write]

Référencé par [getDateDeValidite\(\)](#), [setDateDeValidite\(\)](#), et [setUtilisateurArmoire\(\)](#).

9.30.3.3 m_identifiant

QString UtilisateurArmoire::m_identifiant [read], [write]

Référencé par [getIdentifiant\(\)](#), [setIdentifiant\(\)](#), et [setUtilisateurArmoire\(\)](#).

9.30.3.4 m_idUtilisateur

QString UtilisateurArmoire::m_idUtilisateur [read], [write]

Référencé par [getIdUtilisateur\(\)](#), [setIdUtilisateur\(\)](#), et [setUtilisateurArmoire\(\)](#).

9.30.3.5 m_nom

QString UtilisateurArmoire::m_nom [read], [write]

Référencé par [getNom\(\)](#), [setNom\(\)](#), et [setUtilisateurArmoire\(\)](#).

9.30.3.6 m_nomGroupe

QString UtilisateurArmoire::m_nomGroupe [read], [write]

Référencé par [getNomGroupe\(\)](#), [setNomGroupe\(\)](#), et [setUtilisateurArmoire\(\)](#).

9.30.3.7 m_nomProfil

QString UtilisateurArmoire::m_nomProfil [read], [write]

Référencé par [getNomProfil\(\)](#), [setNomProfil\(\)](#), et [setUtilisateurArmoire\(\)](#).

9.30.3.8 m_prenom

QString UtilisateurArmoire::m_prenom [read], [write]

Référencé par [getPrenom\(\)](#), [setPrenom\(\)](#), et [setUtilisateurArmoire\(\)](#).

La documentation de cette classe a été générée à partir des fichiers suivants :

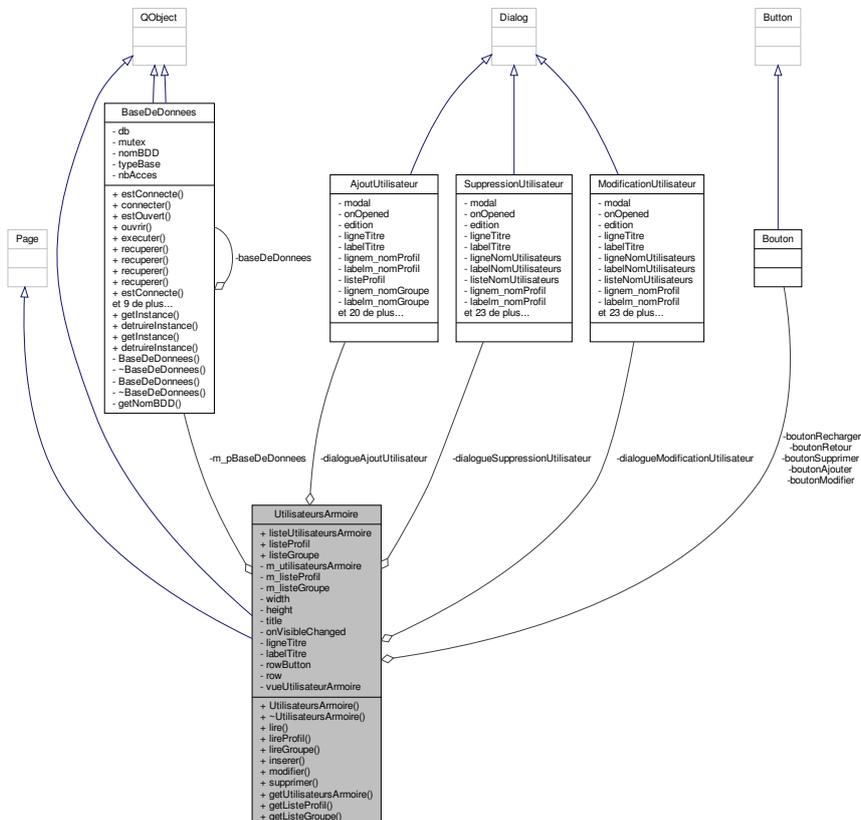
- [utilisateurArmoire.h](#)
- [utilisateurArmoire.cpp](#)

9.31 Référence de la classe UtilisateursArmoire

page des utilisateurs

```
#include <utilisateursArmoire.h>
```

Graphe de collaboration de UtilisateursArmoire :



Signaux

- void [listeUtilisateursArmoireChanged](#) ()
- void [listeProfilChanged](#) ()
- void [listeGroupeChanged](#) ()

Fonctions membres publiques

- [UtilisateursArmoire](#) (QObject *p_parent=nullptr)
- [~UtilisateursArmoire](#) ()
- Q_INVOKABLE bool [lire](#) ()
- Q_INVOKABLE bool [lireProfil](#) ()
- Q_INVOKABLE bool [lireGroupe](#) ()
- Q_INVOKABLE bool [insérer](#) (QString p_idProfil, QString p_idGroupe, QString p_nom, QString p_prenom, QString p_dateDeValidite, QString p_identifiant, QString p_badge)
- Q_INVOKABLE bool [modifier](#) (QString p_idProfil, QString p_idGroupe, QString p_nom, QString p_prenom, QString p_dateDeValidite, QString p_identifiant, QString p_badge, QString p_idUtilisateur)
- Q_INVOKABLE bool [supprimer](#) (QString m_nomProfil, QString m_nomGroupe, QString m_nom, QString m_prenom, QString m_dateDeValidite, QString m_identifiant, QString m_badge, QString p_idUtilisateur)
- QVariant [getUtilisateursArmoire](#) ()
- QVariant [getListeProfil](#) ()
- QVariant [getListeGroupe](#) ()

Propriétés

- QVariant [listeUtilisateursArmoire](#)
- QVariant [listeProfil](#)
- QVariant [listeGroupe](#)

Attributs privés

- QList< QObject * > [m_utilisateursArmoire](#)
- QList< QObject * > [m_listeProfil](#)
- QList< QObject * > [m_listeGroupe](#)
- [BaseDeDonnees](#) * [m_pBaseDeDonnees](#)
- var [width](#)
- var [height](#)
- var [title](#)
- var [onVisibleChanged](#)
- Row [ligneTitre](#)
- *Conteneur du texte labelTitre.*
- Text [labelTitre](#)
- *titre de la boîte de dialogue*
- Row [rowButton](#)
- Bouton [boutonAjouter](#)
- Bouton [boutonModifier](#)
- Bouton [boutonSupprimer](#)
- Bouton [boutonRecharger](#)
- Bouton [boutonRetour](#)
- Row [row](#)
- TableView [vueUtilisateurArmoire](#)
- [AjoutUtilisateur](#) [dialogueAjoutUtilisateur](#)
- [ModificationUtilisateur](#) [dialogueModificationUtilisateur](#)
- [SuppressionUtilisateur](#) [dialogueSuppressionUtilisateur](#)

9.31.1 Description détaillée

La page des utilisateurs.

Auteur

Robin Gauthier

Version

1.1

9.31.2 Documentation des constructeurs et destructeur

9.31.2.1 UtilisateursArmoire()

```
UtilisateursArmoire::UtilisateursArmoire (
    QObject * p_parent = nullptr ) [explicit]
```

Références [BaseDeDonnees : getInstance\(\)](#), et [m_pBaseDeDonnees](#).

```
00007                                     : QObject(p_parent)
00008 {
00009     m_pBaseDeDonnees = BaseDeDonnees::getInstance("MYSQL");
00010 }
```

9.31.2.2 ~UtilisateursArmoire()

```
UtilisateursArmoire::~UtilisateursArmoire ( )
```

Références [BaseDeDonnees : detruireInstance\(\)](#).

```
00013 {
00014     BaseDeDonnees::detruireInstance("MYSQL");
00015 }
```

9.31.3 Documentation des fonctions membres

9.31.3.1 getListeGroupe()

```
QVariant UtilisateursArmoire::getListeGroupe ( )
```

Références [m_listeGroupe](#).

```
00179 {
00180     return QVariant::fromValue(m_listeGroupe);
00181 }
```

9.31.3.2 getListeProfil()

```
QVariant UtilisateursArmoire::getListeProfil ( )
```

Références [m_listeProfil](#).

```
00174 {
00175     return QVariant::fromValue(m_listeProfil);
00176 }
```

9.31.3.3 getUtilisateursArmoire()

```
QVariant UtilisateursArmoire::getUtilisateursArmoire ( )
```

Références [m_utilisateursArmoire](#).

```
00169 {
00170     return QVariant::fromValue(m_utilisateursArmoire);
00171 }
```

9.31.3.4 inserer()

```
bool UtilisateursArmoire::inserer (
    QString p_idProfil,
    QString p_idGroupe,
    QString p_nom,
    QString p_prenom,
    QString p_dateDeValidite,
    QString p_identifiant,
    QString p_badge )
```

Références [BaseDeDonnees : :executer\(\)](#), [lire\(\)](#), et [m_pBaseDeDonnees](#).

```
00124 {
00125     qDebug() << Q_FUNC_INFO << "p_idProfil" << p_idProfil << "p_idGroupe" << p_idGroupe << "p_nom" << p_nom
    << "p_prenom" << p_prenom << "p_dateDeValidite" << p_dateDeValidite << "p_identifiant" << p_identifiant <<
    "p_badge" << p_badge;
00126     //requête INSERT
00127     QString requeteAjoutUtilisateur = "INSERT INTO Utilisateur (idProfil, idGroupe, Nom, Prenom,
    DateValidite, Identifiant, Badge, Email) VALUES ('" + p_idProfil + "', '" + p_idGroupe + "', '" + p_nom + "', '" +
    p_prenom + "', '" + p_dateDeValidite + "', '" + p_identifiant + "', '" + p_badge + "', '')";
00128     qDebug() << Q_FUNC_INFO << "requeteAjoutUtilisateur" << requeteAjoutUtilisateur;
00129     if (m_pBaseDeDonnees->executer(requeteAjoutUtilisateur))
00130     {
00131         // relire la liste
00132         lire();
00133         return true;
00134     }
00135     return false;
00136 }
```

9.31.3.5 lire()

```
bool UtilisateursArmoire::lire ( )
```

Références [listeUtilisateursArmoireChanged\(\)](#), [m_pBaseDeDonnees](#), [m_utilisateursArmoire](#), et [BaseDeDonnees : :recuperer\(\)](#).

Référencé par [inserer\(\)](#), [modifier\(\)](#), et [supprimer\(\)](#).

```
00018 {
00019     QString requete = "SELECT Profil.Nom, Groupe.Nom, Utilisateur.Nom, Utilisateur.Prenom,
    Utilisateur.DateValidite, Utilisateur.Identifiant, Utilisateur.Badge, Utilisateur.idUtilisateur FROM Utilisateur INNER JOIN
    Groupe ON Utilisateur.idGroupe = Groupe.idGroupe INNER JOIN Profil ON Utilisateur.idProfil = Profil.idProfil
";
00020     QVector<QStringList> enregistrementsUtilisateursArmoire;
00021     m_utilisateursArmoire.clear();
00022     if (m_pBaseDeDonnees->recuperer(requete, enregistrementsUtilisateursArmoire))
00023     {
00024         QStringList champsUtilisateurArmoire;
00025         for(int i = 0; i < enregistrementsUtilisateursArmoire.size(); i++)
00026         {
00027             // ...
00028         }
```

```

00029         champsUtilisateurArmoire = enregistrementsUtilisateursArmoire.at(i);
00030         qDebug() << Q_FUNC_INFO << "champsUtilisateurArmoire" << champsUtilisateurArmoire;
00031
00032         UtilisateurArmoire *pUtilisateurArmoire = new
UtilisateurArmoire(
00033             champsUtilisateurArmoire.at(0), champsUtilisateurArmoire.at(1),
00034             champsUtilisateurArmoire.at(2), champsUtilisateurArmoire.at(3),
00035             champsUtilisateurArmoire.at(4), champsUtilisateurArmoire.at(5),
00036             champsUtilisateurArmoire.at(6), champsUtilisateurArmoire.at(7));
00037         m_utilisateursArmoire.append(pUtilisateurArmoire);
00038     }
00039
00040     emit listeUtilisateursArmoireChanged();
00041
00042     return true;
00043 }
00044
00045 return false;
00046 }

```

9.31.3.6 lireGroupe()

```
bool UtilisateursArmoire::lireGroupe ( )
```

Références [listeGroupeChanged\(\)](#), [m_listeGroupe](#), [m_pBaseDeDonnees](#), et [BaseDeDonnees : :recuperer\(\)](#).

```

00074 {
00075     QString requete = "SELECT Groupe.idGroupe, Groupe.Nom FROM Groupe ORDER BY Groupe.idGroupe ASC;";
00076     QVector<QStringList> enregistrementsListeGroupe;
00077
00078     qDebug() << Q_FUNC_INFO << "requete" << requete;
00079     m_listeGroupe.clear();
00080     if (m_pBaseDeDonnees->recuperer(requete, enregistrementsListeGroupe))
00081     {
00082         QStringList champsListeGroupe;
00083         for(int i = 0; i < enregistrementsListeGroupe.size(); i++)
00084         {
00085             champsListeGroupe = enregistrementsListeGroupe.at(i);
00086             qDebug() << Q_FUNC_INFO << "champsListeGroupe" << champsListeGroupe;
00087             Groupe *pGroupe = new Groupe(champsListeGroupe.at(0), champsListeGroupe.at(1));
00088             m_listeGroupe.append(pGroupe);
00089         }
00090         emit listeGroupeChanged();
00091
00092         return true;
00093     }
00094
00095     return false;
00096 }

```

9.31.3.7 lireProfil()

```
bool UtilisateursArmoire::lireProfil ( )
```

Références [listeProfilChanged\(\)](#), [m_listeProfil](#), [m_pBaseDeDonnees](#), et [BaseDeDonnees : :recuperer\(\)](#).

```

00049 {
00050     QString requete = "SELECT Profil.idProfil, Profil.Nom FROM Profil ORDER BY Profil.idProfil ASC;";
00051     QVector<QStringList> enregistrementsListeProfil;
00052
00053     qDebug() << Q_FUNC_INFO << "requete" << requete;
00054     m_listeProfil.clear();
00055     if (m_pBaseDeDonnees->recuperer(requete, enregistrementsListeProfil))
00056     {
00057         QStringList champsListeProfil;
00058         for(int i = 0; i < enregistrementsListeProfil.size(); i++)
00059         {
00060             champsListeProfil = enregistrementsListeProfil.at(i);
00061             qDebug() << Q_FUNC_INFO << "champsListeProfil" << champsListeProfil;
00062             Profil *pProfil = new Profil(champsListeProfil.at(0), champsListeProfil.at(1));
00063             m_listeProfil.append(pProfil);
00064         }
00065         emit listeProfilChanged();
00066
00067         return true;
00068     }
00069
00070     return false;
00071 }

```

9.31.3.8 listeGroupeChanged

```
void UtilisateursArmoire::listeGroupeChanged ( ) [signal]
```

Référencé par [lireGroupe\(\)](#).

9.31.3.9 listeProfilChanged

```
void UtilisateursArmoire::listeProfilChanged ( ) [signal]
```

Référencé par [lireProfil\(\)](#).

9.31.3.10 listeUtilisateursArmoireChanged

```
void UtilisateursArmoire::listeUtilisateursArmoireChanged ( ) [signal]
```

Référencé par [lire\(\)](#).

9.31.3.11 modifier()

```
bool UtilisateursArmoire::modifier (
    QString p_idProfil,
    QString p_idGroupe,
    QString p_nom,
    QString p_prenom,
    QString p_dateDeValidite,
    QString p_identifiant,
    QString p_badge,
    QString p_idUtilisateur )
```

Références [BaseDeDonnees : :executer\(\)](#), [lire\(\)](#), et [m_pBaseDeDonnees](#).

```
00139 {
00140     qDebug() << Q_FUNC_INFO << "p_idProfil" << p_idProfil << "p_idGroupe" << p_idGroupe << "p_nom" << p_nom
    << "p_prenom" << p_prenom << "p_dateDeValidite" << p_dateDeValidite << "p_identifiant" << p_identifiant <<
    "p_badge" << p_badge << "p_idUtilisateur" << p_idUtilisateur;
00141     //requête UPDATE
00142     QString requeteModificationUtilisateur = "UPDATE Utilisateur SET idProfil='" + p_idProfil + "
    ',idGroupe='" + p_idGroupe + "',Nom='" + p_nom + "',Prenom='" + p_prenom + "',DateValidite='" + p_dateDeValidite + "
    ',Identifiant='" + p_identifiant + "',Badge='" + p_badge + "' WHERE idUtilisateur='" + p_idUtilisateur + "'";
00143     qDebug() << Q_FUNC_INFO << "requeteModificationUtilisateur" << requeteModificationUtilisateur;
00144     if (m_pBaseDeDonnees->executer(requeteModificationUtilisateur))
00145     {
00146         // relire la liste
00147         lire();
00148         return true;
00149     }
00150     return false;
00151 }
```

9.31.3.12 supprimer()

```
bool UtilisateursArmoire::supprimer (
    QString m_nomProfil,
    QString m_nomGroupe,
    QString m_nom,
    QString m_prenom,
    QString m_dateDeValidite,
    QString m_identifiant,
    QString m_badge,
    QString p_idUtilisateur )
```

Références [BaseDeDonnees : :executer\(\)](#), [lire\(\)](#), et [m_pBaseDeDonnees](#).

```
00154 {
00155     qDebug() << Q_FUNC_INFO << "p_idProfil" << p_idProfil << "p_idGroupe" << p_idGroupe << "p_nom" << p_nom
    << "p_prenom" << p_prenom << "p_dateDeValidite" << p_dateDeValidite << "p_identifiant" << p_identifiant <<
    "p_badge" << p_badge << "p_idUtilisateur" << p_idUtilisateur;
00156     //requête UPDATE
00157     QString requeteSuppressionUtilisateur = "DELETE FROM Utilisateur WHERE Utilisateur.idUtilisateur = '" +
    p_idUtilisateur + "'";
00158     qDebug() << Q_FUNC_INFO << "requeteSuppressionUtilisateur" << requeteSuppressionUtilisateur;
00159     if (m_pBaseDeDonnees->executer(requeteSuppressionUtilisateur))
00160     {
00161         // relire la liste
00162         lire();
00163         return true;
00164     }
00165     return false;
00166 }
```

9.31.4 Documentation des données membres

9.31.4.1 boutonAjouter

`Bouton` UtilisateursArmoire::boutonAjouter [private]

9.31.4.2 boutonModifier

`Bouton` UtilisateursArmoire::boutonModifier [private]

9.31.4.3 boutonRecharger

`Bouton` UtilisateursArmoire::boutonRecharger [private]

9.31.4.4 boutonRetour

`Bouton` UtilisateursArmoire::boutonRetour [private]

9.31.4.5 boutonSupprimer

`Bouton` UtilisateursArmoire::boutonSupprimer [private]

9.31.4.6 dialogueAjoutUtilisateur

`AjoutUtilisateur` UtilisateursArmoire::dialogueAjoutUtilisateur [private]

9.31.4.7 dialogueModificationUtilisateur

`ModificationUtilisateur` UtilisateursArmoire::dialogueModificationUtilisateur [private]

9.31.4.8 dialogueSuppressionUtilisateur

`SuppressionUtilisateur` UtilisateursArmoire::dialogueSuppressionUtilisateur [private]

9.31.4.9 height

`var` UtilisateursArmoire::height [private]

9.31.4.10 labelTitre

`Text` UtilisateursArmoire::labelTitre [private]

9.31.4.11 ligneTitre

`Row` UtilisateursArmoire::ligneTitre [private]

9.31.4.12 m_listeGroupe

`QList<QObject*>` UtilisateursArmoire::m_listeGroupe [private]

Référencé par [getListeGroupe\(\)](#), et [lireGroupe\(\)](#).

9.31.4.13 m_listeProfil

`QList<QObject*>` UtilisateursArmoire::m_listeProfil [private]

Référencé par [getListeProfil\(\)](#), et [lireProfil\(\)](#).

9.31.4.14 m_pBaseDeDonnees

`BaseDeDonnees*` UtilisateursArmoire::m_pBaseDeDonnees [private]

Référencé par [inserer\(\)](#), [lire\(\)](#), [lireGroupe\(\)](#), [lireProfil\(\)](#), [modifier\(\)](#), [supprimer\(\)](#), et [UtilisateursArmoire\(\)](#).

9.31.4.15 m_utilisateursArmoire

`QList<QObject*>` UtilisateursArmoire::m_utilisateursArmoire [private]

Référencé par [getUtilisateursArmoire\(\)](#), et [lire\(\)](#).

9.31.4.16 onVisibleChanged

`var` UtilisateursArmoire::onVisibleChanged [private]

9.31.4.17 row

`Row` UtilisateursArmoire::row [private]

9.31.4.18 rowButton

`Row` UtilisateursArmoire::rowButton [private]

9.31.4.19 title

`var` UtilisateursArmoire::title [private]

9.31.4.20 vueUtilisateurArmoire

`TableView` UtilisateursArmoire::vueUtilisateurArmoire [private]

9.31.4.21 width

`var` UtilisateursArmoire::width [private]

9.31.5 Documentation des propriétés

9.31.5.1 listeGroupe

```
QVariant UtilisateursArmoire::listeGroupe [read]
```

9.31.5.2 listeProfil

```
QVariant UtilisateursArmoire::listeProfil [read]
```

9.31.5.3 listeUtilisateursArmoire

```
QVariant UtilisateursArmoire::listeUtilisateursArmoire [read]
```

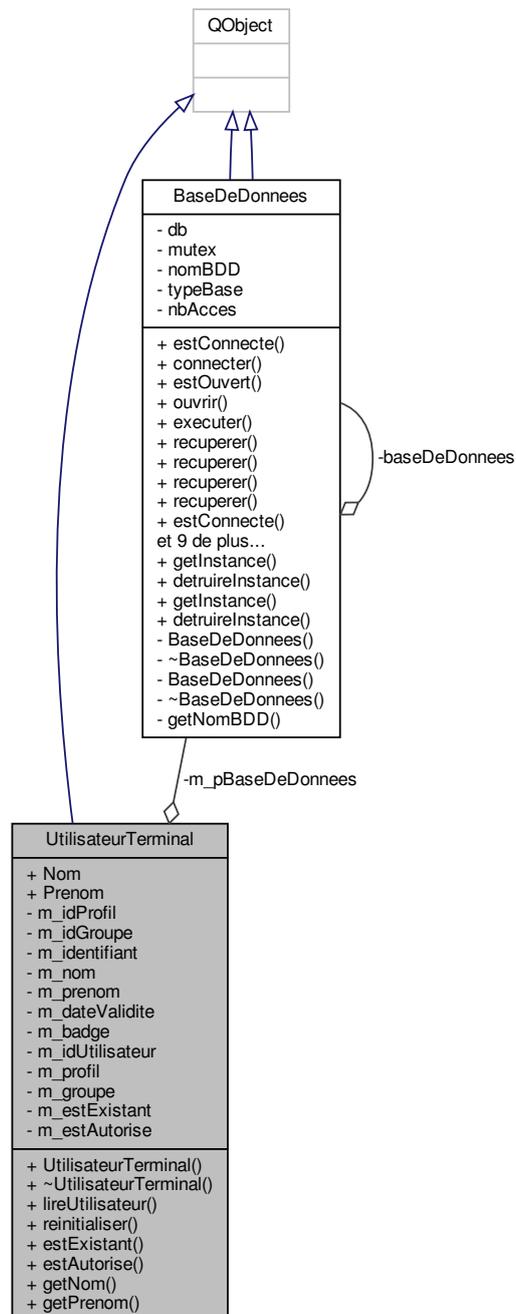
La documentation de cette classe a été générée à partir des fichiers suivants :

- [utilisateursArmoire.h](#)
- [UtilisateursArmoire.qml](#)
- [utilisateursArmoire.cpp](#)

9.32 Référence de la classe UtilisateurTerminal

```
#include <utilisateurTerminal.h>
```

Grphe de collaboration de UtilisateurTerminal :



Signaux

— void `utilisateurUpdated` ()

Fonctions membres publiques

— `UtilisateurTerminal` (`QObject *p_parent=nullptr`)
 — `~UtilisateurTerminal` ()
 — `Q_INVOKABLE` bool `lireUtilisateur` (`QString p_identifiant`, `QString p_password`, bool `p_doForceLogin`)

- Q_INVOKABLE bool [reinitialiser](#) ()
- Q_INVOKABLE bool [estExistant](#) ()
- Q_INVOKABLE bool [estAutorise](#) ()
- QString [getNom](#) () const
- QString [getPrenom](#) () const

Propriétés

- QString [Nom](#)
- QString [Prenom](#)

Attributs privés

- QString [m_idProfil](#)
- QString [m_idGroupe](#)
- QString [m_identifiant](#)
- QString [m_nom](#)
- QString [m_prenom](#)
- QString [m_dateValidite](#)
- QString [m_badge](#)
- QString [m_idUtilisateur](#)
- QString [m_profil](#)
- QString [m_groupe](#)
- [BaseDeDonnees](#) * [m_pBaseDeDonnees](#)
- bool [m_estExistant](#)
- bool [m_estAutorise](#)

9.32.1 Documentation des constructeurs et destructeur

9.32.1.1 UtilisateurTerminal()

```
UtilisateurTerminal::UtilisateurTerminal (
    QObject * p_parent = nullptr )
```

Références [BaseDeDonnees](#) : [getInstance\(\)](#), [m_estAutorise](#), [m_estExistant](#), et [m_pBaseDeDonnees](#).

```
00005                                     : QObject(p_parent)
00006 {
00007     m\_estExistant = false;
00008     m\_estAutorise = false;
00009     m\_pBaseDeDonnees = BaseDeDonnees::getInstance("QMYSQL");
00010 }
```

9.32.1.2 ~UtilisateurTerminal()

```
UtilisateurTerminal::~~UtilisateurTerminal ( )
```

Références [BaseDeDonnees](#) : [destruireInstance\(\)](#).

```
00013 {
00014     BaseDeDonnees::destruireInstance("QMYSQL");
00015 }
```

9.32.2 Documentation des fonctions membres

9.32.2.1 estAutorise()

```
bool UtilisateurTerminal::estAutorise ( )
```

Références [m_estAutorise](#).

```
00128 {  
00129     return m_estAutorise;  
00130 }
```

9.32.2.2 estExistant()

```
bool UtilisateurTerminal::estExistant ( )
```

Références [m_estExistant](#).

```
00123 {  
00124     return m_estExistant;  
00125 }
```

9.32.2.3 getNom()

```
QString UtilisateurTerminal::getNom ( ) const
```

Références [m_nom](#).

```
00133 {  
00134     return m_nom;  
00135 }
```

9.32.2.4 getPrenom()

```
QString UtilisateurTerminal::getPrenom ( ) const
```

Références [m_prenom](#).

```
00138 {  
00139     return m_prenom;  
00140 }
```



```

00084     m_profil = donneesUtilisateur.at(8);
00085     m_groupe = donneesUtilisateur.at(10);
00086
00087     m_estAutorise = true;
00088
00089     emit utilisateurUpdated();
00090
00091     return true;
00092 }
00093 else
00094 {
00095     m_estAutorise = false;
00096
00097     return false;
00098 }
00099
00100 return false;
00101 }

```

9.32.2.6 reinitialiser()

```
bool UtilisateurTerminal::reinitialiser ( )
```

Références [m_badge](#), [m_dateValidite](#), [m_estAutorise](#), [m_estExistant](#), [m_groupe](#), [m_identifiant](#), [m_idGroupe](#), [m_idProfil](#), [m_idUtilisateur](#), [m_nom](#), [m_prenom](#), [m_profil](#), et [utilisateurUpdated\(\)](#).

```

00104 {
00105     qDebug() << Q_FUNC_INFO;
00106     m_idProfil.clear();
00107     m_idGroupe.clear();
00108     m_identifiant.clear();
00109     m_nom.clear();
00110     m_prenom.clear();
00111     m_dateValidite.clear();
00112     m_badge.clear();
00113     m_idUtilisateur.clear();
00114     m_profil.clear();
00115     m_groupe.clear();
00116     m_estExistant = false;
00117     m_estAutorise = false;
00118     emit utilisateurUpdated();
00119     return true;
00120 }

```

9.32.2.7 utilisateurUpdated

```
void UtilisateurTerminal::utilisateurUpdated ( ) [signal]
```

Référencé par [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3 Documentation des données membres

9.32.3.1 m_badge

```
QString UtilisateurTerminal::m_badge [private]
```

Référencé par [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3.2 m_dateValidite

```
QString UtilisateurTerminal::m_dateValidite [private]
```

Référencé par [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3.3 m_estAutorise

```
bool UtilisateurTerminal::m_estAutorise [private]
```

Référencé par [estAutorise\(\)](#), [lireUtilisateur\(\)](#), [reinitialiser\(\)](#), et [UtilisateurTerminal\(\)](#).

9.32.3.4 m_estExistant

```
bool UtilisateurTerminal::m_estExistant [private]
```

Référencé par [estExistant\(\)](#), [lireUtilisateur\(\)](#), [reinitialiser\(\)](#), et [UtilisateurTerminal\(\)](#).

9.32.3.5 m_groupe

```
QString UtilisateurTerminal::m_groupe [private]
```

Référencé par [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3.6 m_identifiant

```
QString UtilisateurTerminal::m_identifiant [private]
```

Référencé par [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3.7 m_idGroupe

```
QString UtilisateurTerminal::m_idGroupe [private]
```

Référencé par [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3.8 m_idProfil

```
QString UtilisateurTerminal::m_idProfil [private]
```

Référencé par [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3.9 m_idUtilisateur

QString UtilisateurTerminal::m_idUtilisateur [private]

Référencé par [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3.10 m_nom

QString UtilisateurTerminal::m_nom [private]

Référencé par [getNom\(\)](#), [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3.11 m_pBaseDeDonnees

BaseDeDonnees* UtilisateurTerminal::m_pBaseDeDonnees [private]

Référencé par [lireUtilisateur\(\)](#), et [UtilisateurTerminal\(\)](#).

9.32.3.12 m_prenom

QString UtilisateurTerminal::m_prenom [private]

Référencé par [getPrenom\(\)](#), [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.3.13 m_profil

QString UtilisateurTerminal::m_profil [private]

Référencé par [lireUtilisateur\(\)](#), et [reinitialiser\(\)](#).

9.32.4 Documentation des propriétés

9.32.4.1 Nom

QString UtilisateurTerminal::Nom

9.32.4.2 Prenom

QString UtilisateurTerminal::Prenom

La documentation de cette classe a été générée à partir des fichiers suivants :

- [utilisateurTerminal.h](#)
- [utilisateurTerminal.cpp](#)

10 Documentation des fichiers

10.1 Référence du fichier administrateur.cpp

```
#include "administrateur.h"
```

10.2 Référence du fichier administrateur.h

```
#include <QObject>  
#include <QString>
```

Classes

— class [Administrateur](#)

10.3 Référence du fichier AjoutArmoire.qml

Définition de la boîte de dialogue [AjoutArmoire](#).

Classes

— class [AjoutArmoire](#)
La boîte de dialogue [AjoutArmoire](#).

10.3.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.4 Référence du fichier AjoutUtilisateur.qml

Définition de la boîte de dialogue [AjoutUtilisateur](#).

Classes

— class [AjoutUtilisateur](#)
La boîte de dialogue [AjoutUtilisateur](#).

10.4.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.5 Référence du fichier armoire.cpp

```
#include "armoire.h"
```

10.6 Référence du fichier armoire.h

```
#include <QObject>
```

Classes

— class [Armoire](#)

10.7 Référence du fichier armoires.cpp

```
#include "armoires.h"  
#include "armoire.h"  
#include "baseDeDonnees.h"
```

10.8 Référence du fichier armoires.h

```
#include <QObject>  
#include <QVariant>
```

Classes

— class [Armoires](#)

10.9 Référence du fichier article.cpp

```
#include "article.h"
```

10.10 Référence du fichier article.h

```
#include <QObject>
#include <QString>
```

Classes

— class [Article](#)

10.11 Référence du fichier AuthenticationBadge.cpp

```
#include "AuthenticationBadge.h"
#include "baseDeDonnees.h"
```

10.12 Référence du fichier AuthenticationBadge.h

```
#include <QObject>
#include <QString>
```

Classes

— class [AuthenticationBadge](#)

10.13 Référence du fichier baseDeDonnees.cpp

Définition de la classe [BaseDeDonnees](#).

```
#include "baseDeDonnees.h"
#include <QDebug>
#include <QMessageBox>
```

10.13.1 Description détaillée

Auteur

Thierry Vaira

Version

1.1

10.14 Référence du fichier baseDeDonnees.cpp

Définition de la classe [BaseDeDonnees](#) pour Android.

```
#include "baseDeDonnees.h"
#include <QDebug>
```

10.14.1 Description détaillée

Auteur

Thierry Vaira

Version

1.1

10.15 Référence du fichier baseDeDonnees.h

Déclaration de la classe [BaseDeDonnees](#).

```
#include <QObject>
#include <QtSql/QtSql>
#include <QSqlDatabase>
#include <QMutex>
#include <QString>
#include "e-stock.h"
```

Classes

— class [BaseDeDonnees](#)

10.15.1 Description détaillée

Auteur

Thierry VAIRA

Version

1.1

10.16 Référence du fichier baseDeDonnees.h

Déclaration de la classe [BaseDeDonnees](#) pour Android.

```
#include <QObject>
#include <QtSql/QtSql>
#include <QSqlDatabase>
#include <QMutex>
#include <QString>
```

Classes

— class [BaseDeDonnees](#)

Macros

- #define `NB_BASEDEDONNEES` 2
- #define `BDD_SQLITE` 0
- #define `BDD_MYSQL` 1

10.16.1 Description détaillée

Auteur

Thierry VAIRA

Version

1.1

10.16.2 Documentation des macros

10.16.2.1 BDD_MYSQL

```
#define BDD_MYSQL 1
```

Référencé par [BaseDeDonnees : :destruireInstance\(\)](#).

10.16.2.2 BDD_SQLITE

```
#define BDD_SQLITE 0
```

Référencé par [BaseDeDonnees : :destruireInstance\(\)](#).

10.16.2.3 NB_BASEDEDONNEES

```
#define NB_BASEDEDONNEES 2
```

10.17 Référence du fichier Bouton.qml

Classes

- class [Bouton](#)

10.18 Référence du fichier casier.cpp

```
#include "casier.h"  
#include <QDebug>
```

10.19 Référence du fichier casier.cpp

```
#include "casier.h"  
#include "baseDeDonnees.h"
```

10.20 Référence du fichier casier.h

```
#include <QtWidgets>
```

Classes

— class [Casier](#)

10.21 Référence du fichier casier.h

```
#include <QObject>
```

Classes

— class [Casier](#)

10.22 Référence du fichier Casier.qml

Définition d'un casier de l'armoire.

Classes

— class [Casier](#)

10.22.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.23 Référence du fichier casiers.cpp

```
#include "casiers.h"  
#include "casier.h"  
#include "baseDeDonnees.h"
```

10.24 Référence du fichier casiers.h

```
#include <QObject>
#include <QTimer>
#include <QVariant>
```

Classes

— class [Casiers](#)

10.25 Référence du fichier Changelog.md

10.26 Référence du fichier e-stock.h

Paramètres généraux de l'application.

```
#include <QDebug>
```

Macros

```
— #define BDD\_SERVEUR "localhost"
— #define BDD\_USERNAME "root"
— #define BDD\_PASSWORD "password"
— #define BDD\_NOMBASE "e-stock"
— #define IHM\_CSS QStringLiteral("background-color : rgb(136, 138, 133);")
— #define DESKTOP
```

10.26.1 Description détaillée

Auteur

Thomas MACHON, Hadrien GIMENEZ et Nathan WAGINAIRE

Version

1.1

10.26.2 Documentation des macros

10.26.2.1 BDD_NOMBASE

```
#define BDD_NOMBASE "e-stock"
```

10.26.2.2 BDD_PASSWORD

```
#define BDD_PASSWORD "password"
```

10.26.2.3 BDD_SERVEUR

```
#define BDD_SERVEUR "localhost"
```

10.26.2.4 BDD_USERNAME

```
#define BDD_USERNAME "root"
```

10.26.2.5 DESKTOP

```
#define DESKTOP
```

10.26.2.6 IHM_CSS

```
#define IHM_CSS QStringLiteral("background-color: rgb(136, 138, 133);")
```

Référencé par [IhmAuthentificationSansBadge : :initialiserFenetre\(\)](#), [IhmGestionArmoire : :initialiserFenetre\(\)](#), [IhmGestionGroupe : :initialiserFenetre\(\)](#), et [IhmAuthentification : :initialiserFenetre\(\)](#).

10.27 Référence du fichier GestionBalances.cpp

```
#include "GestionBalances.h"  
#include <QtSerialPort/QtSerialPort>  
#include <QDebug>
```

10.28 Référence du fichier GestionBalances.h

Déclaration de la classe [GestionBalances](#).

```
#include <QObject>  
#include <QtSerialPort>
```

Classes

- class [GestionBalances](#)
Déclaration de la classe [GestionBalances](#).

Macros

— #define [PORT](#) "/dev/ttyACM0"

10.28.1 Description détaillée

Auteur

Hadrien Gimenez

Version

1.1

10.28.2 Documentation des macros

10.28.2.1 PORT

```
#define PORT "/dev/ttyACM0"
```

Référencé par [GestionBalances](#) : :[GestionBalances\(\)](#).

10.29 Référence du fichier GestionCasier.qml

Définition de la boîte de dialogue [GestionCasier](#).

Classes

— class [GestionCasier](#)
La boîte de dialogue [GestionCasier](#).

10.29.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.30 Référence du fichier groupe.cpp

```
#include "groupe.h"
```

10.31 Référence du fichier groupe.h

```
#include <QObject>
```

Classes

— class [Groupe](#)

10.32 Référence du fichier IhmAuthentication.cpp

Définition de la classe [IhmAuthentication](#).

```
#include "IhmAuthentication.h"  
#include "IhmAuthenticationSansBadge.h"  
#include "ui_IhmAuthentication.h"  
#include "e-stock.h"  
#include "utilisateur.h"  
#include "IhmGestionArmoire.h"  
#include "baseDeDonnees.h"  
#include "AuthenticationBadge.h"  
#include <QMessageBox>
```

10.32.1 Description détaillée

Auteur

Thomas MACHON

Version

0.1

10.33 Référence du fichier IhmAuthentication.h

Déclaration de la classe [IhmAuthentication](#).

```
#include <QtWidgets>
```

Classes

— class [IhmAuthentication](#)
La fenêtre principale de l'application.

Espaces de nommage

— [Ui](#)

10.33.1 Description détaillée

Auteur

Thomas MACHON

Version

1.1

10.34 Référence du fichier IhmAuthenticationSansBadge.cpp

Définition de la classe [IhmAuthenticationSansBadge](#).

```
#include "IhmAuthentication.h"
#include "IhmAuthenticationSansBadge.h"
#include "ui_IhmAuthenticationSansBadge.h"
#include "baseDeDonnees.h"
#include "IhmGestionArmoire.h"
#include "utilisateur.h"
```

10.34.1 Description détaillée

Auteur

Thomas MACHON

Version

0.1

10.35 Référence du fichier IhmAuthenticationSansBadge.h

```
#include <QtWidgets>
```

Classes

— class [IhmAuthenticationSansBadge](#)

Espaces de nommage

— [Ui](#)

10.36 Référence du fichier IhmGestionArmoire.cpp

Définition de la classe [IhmGestionArmoire](#).

```
#include "IhmGestionArmoire.h"  
#include "ui_IhmGestionArmoire.h"  
#include "utilisateur.h"  
#include "IhmGestionGroupe.h"  
#include "IhmAuthentification.h"  
#include "IhmAuthentificationSansBadge.h"  
#include "baseDeDonnees.h"  
#include "casier.h"  
#include "GestionBalances.h"  
#include <QNetworkInterface>
```

10.36.1 Description détaillée

Auteur

Thomas MACHON

Version

0.1

10.37 Référence du fichier IhmGestionArmoire.h

```
#include <QtWidgets>
```

Classes

— class [IhmGestionArmoire](#)

Espaces de nommage

— [Ui](#)

Macros

— `#define` [NB_CASIERS_MAX](#) 8

10.37.1 Documentation des macros

10.37.1.1 NB_CASIERS_MAX

```
#define NB_CASIERS_MAX 8
```

10.38 Référence du fichier IhmGestionBalances.cpp

10.39 Référence du fichier IhmGestionBalances.h

```
#include <QWidget>
```

Classes

— class [IhmGestionBalances](#)

Espaces de nommage

— [Ui](#)

10.40 Référence du fichier IhmGestionGroupe.cpp

```
#include "IhmGestionGroupe.h"  
#include "ui_IhmGestionGroupe.h"  
#include "baseDeDonnees.h"  
#include "utilisateur.h"
```

10.41 Référence du fichier IhmGestionGroupe.h

Déclaration de la classe [IhmGestionGroupe](#).

```
#include <QtWidgets>
```

Classes

— class [IhmGestionGroupe](#)
La fenêtre de gestion des groupes.

Espaces de nommage

— [Ui](#)

10.41.1 Description détaillée

Auteur

Hadrien GIMENEZ

Version

1.1

10.42 Référence du fichier IHMPrincipale.qml

Définition de la fenêtre principale de l'application terminal mobile.

Classes

- class [IHMPrincipale](#)
La fenêtre principale de l'application.

10.42.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.43 Référence du fichier INSTALL.md

10.44 Référence du fichier main.cpp

Programme principal e-stock.

```
#include <QApplication>
#include "IhmAuthentification.h"
```

Fonctions

- int [main](#) (int argc, char *argv[])

10.44.1 Description détaillée

Crée et affiche la fenêtre principale de l'application

Auteur

Thomas MACHON, Hadrien GIMENEZ et Nathan WAGINAIRE

Version

1.1

10.44.2 Documentation des fonctions

10.44.2.1 main()

```
main (
    int argc,
    char * argv[] )
```

Paramètres

<i>argc</i>	
<i>argv[]</i>	

Renvoie

int

```
00023 {
00024     QApplication a(argc, argv);
00025
00026     IhmAuthentication ihmAuthentication;
00027     ihmAuthentication.show();
00028
00029     return a.exec();
00030 }
```

10.45 Référence du fichier main.cpp

Programme principal terminalMobile.

```
#include <QGuiApplication>
#include <QQmlApplicationEngine>
#include <QQmlContext>
#include "baseDeDonnees.h"
#include "utilisateurTerminal.h"
#include "mouvements.h"
#include "utilisateursArmoire.h"
#include "utilisateurArmoire.h"
#include "armoires.h"
#include "casiers.h"
```

Fonctions

— int [main](#) (int argc, char *argv[])

10.45.1 Description détaillée

Crée et affiche la fenêtre principale de l'application

Auteur

Robin Gauthier

Version

1.1

10.45.2 Documentation des fonctions

10.45.2.1 main()

```
int main (
    int argc,
    char * argv[] )
```

Références [BaseDeDonnees : :destruireInstance\(\)](#), [BaseDeDonnees : :getInstance\(\)](#), et [BaseDeDonnees : :ouvrir\(\)](#).

```
00031 {
00032     QCoreApplication::setApplicationName("e-stock mobile");
00033     QCoreApplication::setAttribute(Qt::AA_EnableHighDpiScaling);
00034
00035     QGuiApplication app(argc, argv);
00036
00037     QQmlApplicationEngine engine;
00038
00039     BaseDeDonnees *g_pBddTerminal = BaseDeDonnees::getInstance("
00040     QSQLITE");
00041     g_pBddTerminal->ouvrir("estock.sqlite");
00042     engine.rootContext()->setContextProperty("g_pBddTerminal", g_pBddTerminal);
00043
00044     Armoires *g_pArmoires = new Armoires;
00045     engine.rootContext()->setContextProperty("g_pArmoires", g_pArmoires);
00046
00047     BaseDeDonnees *g_pBddArmoire = BaseDeDonnees::getInstance("
00048     QMYSQL");
00049     engine.rootContext()->setContextProperty("g_pBddArmoire", g_pBddArmoire);
00050
00051     UtilisateurTerminal *g_pUtilisateurTerminal = new
00052     UtilisateurTerminal;
00053     engine.rootContext()->setContextProperty("g_pUtilisateurTerminal", g_pUtilisateurTerminal);
00054
00055     Mouvements *g_pMouvements = new Mouvements;
00056     engine.rootContext()->setContextProperty("g_pMouvements", g_pMouvements);
00057
00058     UtilisateursArmoire *g_pUtilisateursArmoire = new
00059     UtilisateursArmoire;
00060     engine.rootContext()->setContextProperty("g_pUtilisateursArmoire", g_pUtilisateursArmoire);
00061
00062     Casiers *g_pCasiers = new Casiers;
00063     engine.rootContext()->setContextProperty("g_pCasiers", g_pCasiers);
00064
00065     engine.load(QUrl(QStringLiteral("qrc:/IHMPPrincipale.qml")));
00066     if (engine.rootObjects().isEmpty())
00067         return -1;
00068
00069     int resultat = app.exec();
00070
00071     BaseDeDonnees::destruireInstance("QMYSQL");
00072     BaseDeDonnees::destruireInstance("QSQLITE");
00073
00074     return resultat;
00075 }
```

10.46 Référence du fichier MenuArmoire.qml

Définition de la page de gestion d'une armoire.

Classes

- class [MenuArmoire](#)
page de la gestion de l'Armoire

10.46.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.47 Référence du fichier ModificationUtilisateur.qml

Définition de la boîte de dialogue [ModificationUtilisateur](#).

Classes

- class [ModificationUtilisateur](#)
La boîte de dialogue [ModificationUtilisateur](#).

10.47.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.48 Référence du fichier mouvement.cpp

```
#include "mouvement.h"
```

10.49 Référence du fichier mouvement.h

```
#include <QObject>
```

Classes

- class [Mouvement](#)

10.50 Référence du fichier mouvements.cpp

```
#include "mouvements.h"  
#include "mouvement.h"  
#include "baseDeDonnees.h"
```

10.51 Référence du fichier mouvements.h

```
#include <QObject>  
#include <QVariant>
```

Classes

— class [Mouvements](#)

10.52 Référence du fichier MouvementsArmoire.qml

Définition de la page des mouvements.

Classes

— class [MouvementsArmoire](#)
page des mouvements d'une armoire

10.52.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.53 Référence du fichier profil.cpp

```
#include "profil.h"
```

10.54 Référence du fichier profil.h

```
#include <QObject>
```

Classes

— class [Profil](#)

10.55 Référence du fichier README.md

10.56 Référence du fichier StockArmoire.qml

Définition de la page de gestion d'une armoire.

Classes

— class [StockArmoire](#)
page de la gestion de l'Armoire

10.56.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.57 Référence du fichier SuppressionUtilisateur.qml

Définition de la boîte de dialogue [SuppressionUtilisateur](#).

Classes

- class [SuppressionUtilisateur](#)
La boîte de dialogue [SuppressionUtilisateur](#).

10.57.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.58 Référence du fichier utilisateur.cpp

```
#include "utilisateur.h"  
#include "baseDeDonnees.h"
```

10.59 Référence du fichier utilisateur.h

```
#include <QObject>  
#include <QString>
```

Classes

- class [Utilisateur](#)

10.60 Référence du fichier utilisateurArmoire.cpp

```
#include "utilisateurArmoire.h"
```

10.61 Référence du fichier utilisateurArmoire.h

```
#include <QObject>
```

Classes

— class [UtilisateurArmoire](#)

10.62 Référence du fichier utilisateursArmoire.cpp

```
#include "utilisateursArmoire.h"  
#include "utilisateurArmoire.h"  
#include "profil.h"  
#include "groupe.h"  
#include "baseDeDonnees.h"
```

10.63 Référence du fichier utilisateursArmoire.h

```
#include <QObject>  
#include <QVariant>
```

Classes

— class [UtilisateursArmoire](#)
page des utilisateurs

10.64 Référence du fichier UtilisateursArmoire.qml

Définition de la page des utilisateurs.

Classes

— class [UtilisateursArmoire](#)
page des utilisateurs

10.64.1 Description détaillée

Auteur

Robin Gauthier

Version

1.1

10.65 Référence du fichier utilisateurTerminal.cpp

```
#include "utilisateurTerminal.h"  
#include "baseDeDonnees.h"  
#include <QDebug>
```

10.66 Référence du fichier utilisateurTerminal.h

```
#include <QObject>
```

Classes

— class [UtilisateurTerminal](#)

Index

- ~Administrateur
 - Administrateur, [13](#)
- ~Armoires
 - Armoires, [28](#)
- ~Article
 - Article, [31](#)
- ~AuthenticationBadge
 - AuthenticationBadge, [35](#)
- ~BaseDeDonnees
 - BaseDeDonnees, [40, 41](#)
- ~Casier
 - Casier, [53](#)
- ~Casiers
 - Casiers, [62](#)
- ~GestionBalances
 - GestionBalances, [67](#)
- ~IhmAuthentication
 - IhmAuthentication, [77](#)
- ~IhmAuthenticationSansBadge
 - IhmAuthenticationSansBadge, [84](#)
- ~IhmGestionArmoire
 - IhmGestionArmoire, [92](#)
- ~IhmGestionGroupe
 - IhmGestionGroupe, [104](#)
- ~Mouvements
 - Mouvements, [132](#)
- ~Utilisateur
 - Utilisateur, [152](#)
- ~UtilisateurTerminal
 - UtilisateurTerminal, [178](#)
- ~UtilisateursArmoire
 - UtilisateursArmoire, [169](#)
- actionChanged
 - Mouvement, [125](#)
- actionner
 - Casier, [54](#)
- actualiser
 - Casiers, [63](#)
- administrateur
 - IhmAuthentication, [80](#)
- Administrateur, [12](#)
 - ~Administrateur, [13](#)
 - Administrateur, [13](#)
 - getNom, [14](#)
 - getPrenom, [14](#)
 - identifiantConnexion, [15](#)
 - mail, [15](#)
 - motDePasse, [15](#)
 - nom, [15](#)
 - prenom, [15](#)
 - setNom, [14](#)
 - setPrenom, [14](#)
- administrateur.cpp, [184](#)
- administrateur.h, [184](#)
- adresseIPArmoire
 - IHMPrincipale, [114](#)
- adresseIPChanged
 - Armoire, [25](#)
- afficherAuthenticationNonValide
 - IhmAuthentication, [77](#)
- afficherCasiers
 - IhmGestionArmoire, [92](#)
 - StockArmoire, [141](#)
- afficherListerUtilisateursGroupe
 - IhmGestionGroupe, [104](#)
- AjoutArmoire, [16](#)
 - boutonAjouter, [17](#)
 - boutonAnnuler, [17](#)
 - boutons, [17](#)
 - edition, [17](#)
 - footer, [17](#)
 - labelAdresseIP, [17](#)
 - labelNomArmoire, [17](#)
 - labelTitre, [18](#)
 - ligneAdresseIP, [18](#)
 - ligneNomArmoire, [18](#)
 - ligneTitre, [18](#)
 - modal, [18](#)
 - saisieAdresseIP, [18](#)
 - saisieNomArmoire, [18](#)
- AjoutArmoire.qml, [184](#)
- AjoutUtilisateur, [19](#)
 - boutonAjouter, [20](#)
 - boutonAnnuler, [20](#)
 - boutons, [20](#)
 - edition, [20](#)
 - footer, [21](#)
 - labelTitre, [21](#)
 - labelm_badge, [21](#)
 - labelm_dateDeValidite, [21](#)
 - labelm_identifiant, [21](#)
 - labelm_nom, [21](#)
 - labelm_nomGroupe, [21](#)
 - labelm_nomProfil, [21](#)
 - labelm_prenom, [21](#)
 - ligneTitre, [22](#)
 - lignem_badge, [22](#)
 - lignem_dateDeValidite, [22](#)
 - lignem_identifiant, [22](#)
 - lignem_nom, [22](#)
 - lignem_nomGroupe, [22](#)
 - lignem_nomProfil, [22](#)
 - lignem_prenom, [22](#)
 - listeGroupe, [22](#)
 - listeProfil, [23](#)
 - modal, [23](#)
 - onOpened, [23](#)
 - saisiem_badge, [23](#)
 - saisiem_dateDeValidite, [23](#)
 - saisiem_identifiant, [23](#)
 - saisiem_nom, [23](#)
 - saisiem_prenom, [23](#)
- AjoutUtilisateur.qml, [184](#)
- ajouterGroupe
 - IhmGestionGroupe, [104](#)

- Armoire, 24
 - adresseIPChanged, 25
 - Armoire, 24
 - getAdresseIP, 25
 - getNom, 25
 - m_adresseIP, 26
 - m_nom, 26
 - nomChanged, 25
 - setAdresseIP, 25
 - setArmoire, 26
 - setNom, 26
- armoire
 - MenuArmoire, 118
 - StockArmoire, 143
- armoire.cpp, 185
- armoire.h, 185
- Armoires, 27
 - ~Armoires, 28
 - Armoires, 28
 - getArmoires, 28
 - inserer, 29
 - lire, 29
 - listeArmoires, 30
 - listeArmoiresChanged, 29
 - m_armoires, 30
 - m_pBaseDeDonnees, 30
 - supprimer, 30
- armoires.cpp, 185
- armoires.h, 185
- arreterRafrisissement
 - Casiers, 63
- Article, 31
 - ~Article, 31
 - Article, 31
 - horodatage, 32
 - nomArticle, 33
 - quantite, 33
 - rechercherArticle, 32
 - recupererArticle, 32
 - restituerArticle, 32
 - visualiserLesMouvements, 32
- article
 - IhmGestionArmoire, 99
- article.cpp, 185
- article.h, 186
- articlesTrouves
 - IhmGestionArmoire, 99
- AuthenticationBadge, 33
 - ~AuthenticationBadge, 35
 - AuthenticationBadge, 35
 - autorise, 38
 - bdd, 38
 - dateValidite, 38
 - estAutorise, 35
 - estValide, 35
 - getDateValidite, 36
 - getIdUtilisateur, 36
 - getIdentifiantBadge, 36
 - idUtilisateur, 38
 - identifiantBadge, 38
 - lireBadge, 36
 - valide, 38
- authenticationBadge
 - IhmAuthentication, 80
- AuthenticationBadge.cpp, 186
- AuthenticationBadge.h, 186
- authentifieurSansBadge
 - IhmAuthentication, 77
- autorise
 - AuthenticationBadge, 38
 - Casier, 59
 - IhmAuthenticationSansBadge, 88
- autoriserSansBadge
 - IhmAuthentication, 78
- BDD_MYSQL
 - terminalmobile/baseDeDonnees.h, 188
- BDD_NOMBASE
 - e-stock.h, 190
- BDD_PASSWORD
 - e-stock.h, 190
- BDD_SERVEUR
 - e-stock.h, 191
- BDD_SQLITE
 - terminalmobile/baseDeDonnees.h, 188
- BDD_USERNAME
 - e-stock.h, 191
- badge
 - Utilisateur, 157
- badgeChanged
 - UtilisateurArmoire, 161
- BaseDeDonnees, 39
 - ~BaseDeDonnees, 40, 41
 - BaseDeDonnees, 40, 41
 - baseDeDonnees, 50
 - connecter, 41, 42
 - db, 50
 - deconnecter, 42
 - destruireInstance, 42
 - estConnecte, 43
 - estOuvert, 44
 - executer, 44
 - getInstance, 45
 - getNomBDD, 45
 - mutex, 51
 - nbAcces, 51
 - nomBDD, 51
 - ouvrir, 45, 46
 - recuperer, 46–50
 - typeBase, 51
- baseDeDonnees
 - BaseDeDonnees, 50
- baseDeDonnees.cpp, 186
- baseDeDonnees.h, 187
- bdd
 - AuthenticationBadge, 38
 - IhmAuthentication, 80
 - IhmAuthenticationSansBadge, 88
 - IhmGestionArmoire, 99
 - IhmGestionGroupe, 109
 - Utilisateur, 157
- bddArmoireBaseName

- IHMPrincipale, 115
- bddArmoirePassword
 - IHMPrincipale, 115
- bddArmoireUsername
 - IHMPrincipale, 115
- Bouton, 51
- Bouton.qml, 188
- boutonAjouter
 - AjoutArmoire, 17
 - AjoutUtilisateur, 20
 - GestionCasier, 71
 - UtilisateursArmoire, 173
- boutonAjouterArmoire
 - IHMPrincipale, 111
- boutonAnnuler
 - AjoutArmoire, 17
 - AjoutUtilisateur, 20
 - GestionCasier, 71
 - ModificationUtilisateur, 120
 - SuppressionUtilisateur, 145
- boutonConnecterArmoire
 - IHMPrincipale, 112
- boutonModifier
 - ModificationUtilisateur, 120
 - UtilisateursArmoire, 173
- boutonQuitterStockArmoire
 - StockArmoire, 141
- boutonRecharger
 - MouvementsArmoire, 135
 - UtilisateursArmoire, 173
- boutonRetour
 - MouvementsArmoire, 135
 - UtilisateursArmoire, 173
- boutonSupprimer
 - IHMPrincipale, 112
 - SuppressionUtilisateur, 145
 - UtilisateursArmoire, 173
- boutons
 - AjoutArmoire, 17
 - AjoutUtilisateur, 20
 - GestionCasier, 71
 - ModificationUtilisateur, 120
 - SuppressionUtilisateur, 145
- button
 - MenuArmoire, 116
- button1
 - MenuArmoire, 116
- button4
 - MenuArmoire, 116
- button5
 - MenuArmoire, 116
- Casier, 52
 - ~Casier, 53
 - actionner, 54
 - autorise, 59
 - Casier, 53, 54
 - clicked, 54
 - color, 59
 - disponible, 60
 - disponibleChanged, 55
 - estAutorise, 55
 - estOuvert, 55
 - fermer, 55
 - getDisponible, 55
 - getNomArticle, 56
 - getNumeroCasier, 56
 - getQuantite, 56
 - height, 59
 - mousePressEvent, 56
 - nomArticle, 60
 - nomArticleChanged, 57
 - nomCasier, 60
 - numero, 59
 - numeroCasier, 60
 - numeroCasierChanged, 57
 - ouvert, 59
 - ouvrir, 57
 - quantite, 60
 - quantiteChanged, 57
 - radius, 59
 - setCasier, 57
 - setDisponible, 58
 - setNomArticle, 58
 - setNumeroCasier, 58
 - setQuantite, 58
 - visible, 59
 - width, 60
- casier.cpp, 188, 189
- casier.h, 189
- Casier.qml, 189
- casierChanged
 - Mouvement, 125
- Casiers, 61
 - ~Casiers, 62
 - actualiser, 63
 - arreterRafraichissement, 63
 - Casiers, 62
 - casiers, 64
 - getCasiers, 63
 - getNbCasiers, 63
 - lire, 63
 - listeCasiers, 65
 - listeCasiersChanged, 64
 - m_adresseIPArmoire, 65
 - m_nbCasiers, 65
 - m_pBaseDeDonnees, 65
 - m_timer, 65
 - nbCasiersChanged, 64
- casiers
 - Casiers, 64
 - IhmGestionArmoire, 99
- casiers.cpp, 189
- casiers.h, 190
- Changelog.md, 190
- clicked
 - Casier, 54
- colonneBoutons
 - StockArmoire, 141
- colonneCasier1
 - StockArmoire, 142
- colonneCasier2

- StockArmoire, 142
- colonneCasiersDroite
 - StockArmoire, 142
- colonneCasiersGauche
 - StockArmoire, 142
- colonneChoixArmoire
 - IHMPrincipale, 112
- color
 - Casier, 59
- columnLayout
 - MenuArmoire, 117
- connecter
 - BaseDeDonnees, 41, 42
- conteneurCasiers
 - StockArmoire, 142
- contenuChanged
 - Mouvement, 125
- DESKTOP
 - e-stock.h, 191
- dateDeValiditeChanged
 - UtilisateurArmoire, 161
- dateValidite
 - AuthenticationBadge, 38
 - IhmAuthenticationSansBadge, 88
 - Utilisateur, 157
- db
 - BaseDeDonnees, 50
- debug
 - IHMPrincipale, 115
- deconnecter
 - BaseDeDonnees, 42
- detruireInstance
 - BaseDeDonnees, 42
- dialogueAjoutArmoire
 - IHMPrincipale, 112
- dialogueAjoutUtilisateur
 - UtilisateursArmoire, 174
- dialogueGestionCasier
 - StockArmoire, 142
- dialogueModificationUtilisateur
 - UtilisateursArmoire, 174
- dialogueSuppressionUtilisateur
 - UtilisateursArmoire, 174
- disponible
 - Casier, 60
- disponibleChanged
 - Casier, 55
- donnees
 - GestionBalances, 69
- e-stock.h, 190
 - BDD_NOMBASE, 190
 - BDD_PASSWORD, 190
 - BDD_SERVEUR, 191
 - BDD_USERNAME, 191
 - DESKTOP, 191
 - IHM_CSS, 191
- edition
 - AjoutArmoire, 17
 - AjoutUtilisateur, 20
- GestionCasier, 71
 - ModificationUtilisateur, 120
 - SuppressionUtilisateur, 145
- effacerLaListeArticle
 - IhmGestionArmoire, 93
- effacerLaSelectionArticle
 - IhmGestionArmoire, 93
- effacerMessage
 - IhmAuthentication, 78
 - IhmAuthenticationSansBadge, 84
- effacerMessageNumeroCasier
 - IhmGestionArmoire, 93
- effacerRechercheArticle
 - IhmGestionArmoire, 94
- email
 - Utilisateur, 157
- estAdministrateur
 - Utilisateur, 152
- estAutorise
 - AuthenticationBadge, 35
 - Casier, 55
 - IhmAuthenticationSansBadge, 84
 - UtilisateurTerminal, 178
- estConnecte
 - BaseDeDonnees, 43
- estExistant
 - UtilisateurTerminal, 179
- estGestionnaire
 - Utilisateur, 152
- estOuvert
 - BaseDeDonnees, 44
 - Casier, 55
- estUtilisateur
 - Utilisateur, 153
- estValide
 - AuthenticationBadge, 35
 - IhmAuthenticationSansBadge, 85
- executer
 - BaseDeDonnees, 44
- fermer
 - Casier, 55
 - IhmAuthenticationSansBadge, 85
 - IhmGestionArmoire, 94
 - IhmGestionGroupe, 105
- fermerClavierVirtuel
 - IhmGestionArmoire, 94
- fermerPort
 - GestionBalances, 67
- footer
 - AjoutArmoire, 17
 - AjoutUtilisateur, 21
 - GestionCasier, 71
 - IHMPrincipale, 112
 - ModificationUtilisateur, 120
 - SuppressionUtilisateur, 146
- gererActions
 - IhmGestionGroupe, 105
- gererGroupe
 - IhmGestionArmoire, 94

- GestionBalances, 66
 - ~GestionBalances, 67
 - donnees, 69
 - fermerPort, 67
 - GestionBalances, 67
 - lirePort, 68
 - ouvrirPort, 68
 - parametrerPort, 68
 - port, 69
 - seConnecter, 69
 - seDeconnecter, 69
- gestionBalances
 - IhmGestionArmoire, 99
- GestionBalances.cpp, 191
- GestionBalances.h, 191
 - PORT, 192
- GestionCasier, 70
 - boutonAjouter, 71
 - boutonAnnuler, 71
 - boutons, 71
 - edition, 71
 - footer, 71
 - labelTitre, 71
 - ligneTitre, 71
 - modal, 71
- GestionCasier.qml, 192
- getAction
 - Mouvement, 126
- getAdresselP
 - Armoire, 25
- getArmoires
 - Armoires, 28
- getBadge
 - UtilisateurArmoire, 161
- getCasier
 - IhmGestionArmoire, 95
 - Mouvement, 126
- getCasiers
 - Casiers, 63
- getContenu
 - Mouvement, 126
- getDateDeValidite
 - UtilisateurArmoire, 161
- getDateValidite
 - AuthenticationBadge, 36
- getDisponible
 - Casier, 55
- getGroupe
 - Utilisateur, 153
- getHorodatage
 - Mouvement, 126
- getId
 - Groupe, 73
 - Profil, 138
- getIdGroupe
 - IhmGestionGroupe, 106
 - Utilisateur, 153
- getIdUtilisateur
 - AuthenticationBadge, 36
 - IhmAuthenticationSansBadge, 86
 - Utilisateur, 153
- UtilisateurArmoire, 161
- getIdentifiant
 - UtilisateurArmoire, 161
- getIdentifiantBadge
 - AuthenticationBadge, 36
- getInstance
 - BaseDeDonnees, 45
- getListeGroupe
 - UtilisateursArmoire, 169
- getListeProfil
 - UtilisateursArmoire, 169
- getMouvements
 - Mouvements, 132
- getNbCasiers
 - Casiers, 63
- getNom
 - Administrateur, 14
 - Armoire, 25
 - Groupe, 73
 - Profil, 138
 - Utilisateur, 154
 - UtilisateurArmoire, 162
 - UtilisateurTerminal, 179
- getNomArticle
 - Casier, 56
- getNomBDD
 - BaseDeDonnees, 45
- getNomGroupe
 - UtilisateurArmoire, 162
- getNomProfil
 - UtilisateurArmoire, 162
- getNomUtilisateur
 - IhmGestionGroupe, 106
- getNumeroCasier
 - Casier, 56
- getPrenom
 - Administrateur, 14
 - Utilisateur, 154
 - UtilisateurArmoire, 162
 - UtilisateurTerminal, 179
- getProfil
 - Utilisateur, 154
- getQuantite
 - Casier, 56
 - Mouvement, 126
- getUtilisateur
 - IhmGestionArmoire, 95
 - Mouvement, 127
- getUtilisateursArmoire
 - UtilisateursArmoire, 169
- Groupe, 72
 - getId, 73
 - getNom, 73
 - Groupe, 73
 - idChanged, 73
 - m_id, 74
 - m_nom, 74
 - nomChanged, 73
 - setId, 73
 - setNom, 74
- groupe

- Utilisateur, 157
- groupe.cpp, 192
- groupe.h, 193
- groupes
 - IhmGestionGroupe, 109
- header
 - IHMPrincipale, 112
- height
 - Casier, 59
 - IHMPrincipale, 112
 - MenuArmoire, 117
 - MouvementsArmoire, 135
 - StockArmoire, 142
 - UtilisateursArmoire, 174
- horodatage
 - Article, 32
- horodatageChanged
 - Mouvement, 127
- IHM_CSS
 - e-stock.h, 191
- IHMPrincipale, 110
 - adressesIPArmoire, 114
 - bddArmoireBaseName, 115
 - bddArmoirePassword, 115
 - bddArmoireUsername, 115
 - boutonAjouterArmoire, 111
 - boutonConnecterArmoire, 112
 - boutonSupprimer, 112
 - colonneChoixArmoire, 112
 - debug, 115
 - dialogueAjoutArmoire, 112
 - footer, 112
 - header, 112
 - height, 112
 - indicateur, 112
 - labelChoixArmoire, 112
 - labelMotDePasse, 113
 - labelUtilisateur, 113
 - ligneChoixArmoire, 113
 - ligneConnexion, 113
 - ligneIndicateur, 113
 - ligneMotDePasse, 113
 - ligneUtilisateur, 113
 - listeChoixArmoire, 113
 - logo, 113
 - logoProjet, 114
 - pageMenuArmoire, 114
 - saisieMotDePasse, 114
 - saisieUtilisateur, 114
 - title, 114
 - visible, 114
 - width, 114
- IHMPrincipale.qml, 197
- INSTALL.md, 197
- idChanged
 - Groupe, 73
 - Profil, 138
- idGroupe
 - Utilisateur, 158
- idProfil
 - Utilisateur, 158
- idUtilisateur
 - AuthenticationBadge, 38
 - IhmAuthenticationSansBadge, 88
 - Utilisateur, 158
- idUtilisateurChanged
 - UtilisateurArmoire, 163
- identifiant
 - IhmAuthenticationSansBadge, 88
 - Utilisateur, 157
- identifiantBadge
 - AuthenticationBadge, 38
- identifiantChanged
 - UtilisateurArmoire, 162
- identifiantConnexion
 - Administrateur, 15
- ihm-estock/main.cpp
 - main, 197
- IhmAuthentication, 75
 - ~IhmAuthentication, 77
 - administrateur, 80
 - afficherAuthenticationNonValide, 77
 - authenticationBadge, 80
 - authentifieurSansBadge, 77
 - autoriserSansBadge, 78
 - bdd, 80
 - effacerMessage, 78
 - IhmAuthentication, 76
 - ihmAuthenticationSansBadge, 80
 - ihmGestionArmoire, 80
 - initialiserFenetre, 78
 - passerSansBadge, 79
 - reinitialiserSaisieBadge, 79
 - ui, 81
 - utilisateur, 81
 - validerAcces, 79
- IhmAuthentication.cpp, 193
- IhmAuthentication.h, 193
- IhmAuthenticationSansBadge, 81
 - ~IhmAuthenticationSansBadge, 84
 - autorise, 88
 - bdd, 88
 - dateValidite, 88
 - effacerMessage, 84
 - estAutorise, 84
 - estValide, 85
 - fermer, 85
 - getIdUtilisateur, 86
 - idUtilisateur, 88
 - identifiant, 88
 - IhmAuthenticationSansBadge, 83
 - initialiserClavierVirtuel, 86
 - initialiserFenetre, 86
 - motDePasse, 89
 - showEvent, 87
 - supprimerClavierVirtuel, 87
 - ui, 89
 - utilisateurAutorise, 87
 - valide, 89
 - validerAcces, 87

- ihmAuthenticationSansBadge
 - IhmAuthentication, 80
- IhmAuthenticationSansBadge.cpp, 194
- IhmAuthenticationSansBadge.h, 194
- IhmGestionArmoire, 89
 - ~IhmGestionArmoire, 92
 - afficherCasiers, 92
 - article, 99
 - articlesTrouves, 99
 - bdd, 99
 - casiers, 99
 - effacerLaListeArticle, 93
 - effacerLaSelectionArticle, 93
 - effacerMessageNumeroCasier, 93
 - effacerRechercheArticle, 94
 - fermer, 94
 - fermerClavierVirtuel, 94
 - genererGroupe, 94
 - gestionBalances, 99
 - getCasier, 95
 - getUtilisateur, 95
 - IhmGestionArmoire, 91
 - ihmGestionGroupe, 99
 - initialiserClavierVirtuel, 95
 - initialiserFenetre, 95
 - lireAdresseIP, 96
 - rechercherArticle, 96
 - selectionnerArticle, 97
 - selectionnerCasier, 97
 - setUtilisateur, 98
 - showEvent, 98
 - supprimerClavierVirtuel, 99
 - ui, 100
 - utilisateur, 100
- ihmGestionArmoire
 - IhmAuthentication, 80
- IhmGestionArmoire.cpp, 195
- IhmGestionArmoire.h, 195
 - NB_CASIER_MAX, 195
- IhmGestionBalances, 100
 - ui, 101
- IhmGestionBalances.cpp, 196
- IhmGestionBalances.h, 196
- IhmGestionGroupe, 101
 - ~IhmGestionGroupe, 104
 - afficherListerUtilisateursGroupe, 104
 - ajouterGroupe, 104
 - bdd, 109
 - fermer, 105
 - genererActions, 105
 - getIdGroupe, 106
 - getNomUtilisateur, 106
 - groupes, 109
 - IhmGestionGroupe, 103
 - initialiserFenetre, 106
 - listerGroupe, 106
 - modifierGroupe, 107
 - selectionnerGroupe, 107
 - setUtilisateur, 108
 - showEvent, 108
 - supprimerGroupe, 108
 - ui, 109
 - utilisateur, 109
 - utilisateurs, 110
- ihmGestionGroupe
 - IhmGestionArmoire, 99
- IhmGestionGroupe.cpp, 196
- IhmGestionGroupe.h, 196
- indicateur
 - IHMPrincipale, 112
- initialiserClavierVirtuel
 - IhmAuthenticationSansBadge, 86
 - IhmGestionArmoire, 95
- initialiserFenetre
 - IhmAuthentication, 78
 - IhmAuthenticationSansBadge, 86
 - IhmGestionArmoire, 95
 - IhmGestionGroupe, 106
- insérer
 - Armoires, 29
 - UtilisateursArmoire, 170
- labelAdresseIP
 - AjoutArmoire, 17
- labelArmoire
 - MenuArmoire, 117
- labelChoixArmoire
 - IHMPrincipale, 112
- labelMotDePasse
 - IHMPrincipale, 113
- labelNomArmoire
 - AjoutArmoire, 17
- labelNomUtilisateurs
 - ModificationUtilisateur, 121
 - SuppressionUtilisateur, 146
- labelTitre
 - AjoutArmoire, 18
 - AjoutUtilisateur, 21
 - GestionCasier, 71
 - ModificationUtilisateur, 121
 - MouvementsArmoire, 135
 - SuppressionUtilisateur, 147
 - UtilisateursArmoire, 174
- labelUtilisateur
 - IHMPrincipale, 113
 - StockArmoire, 142
- labelm_badge
 - AjoutUtilisateur, 21
 - ModificationUtilisateur, 120
 - SuppressionUtilisateur, 146
- labelm_dateDeValidite
 - AjoutUtilisateur, 21
 - ModificationUtilisateur, 120
 - SuppressionUtilisateur, 146
- labelm_identifiant
 - AjoutUtilisateur, 21
 - ModificationUtilisateur, 120
 - SuppressionUtilisateur, 146
- labelm_nom
 - AjoutUtilisateur, 21
 - ModificationUtilisateur, 120
 - SuppressionUtilisateur, 146

labelm_nomGroupe
 AjoutUtilisateur, 21
 ModificationUtilisateur, 121
 SuppressionUtilisateur, 146
 labelm_nomProfil
 AjoutUtilisateur, 21
 ModificationUtilisateur, 121
 SuppressionUtilisateur, 146
 labelm_prenom
 AjoutUtilisateur, 21
 ModificationUtilisateur, 121
 SuppressionUtilisateur, 146
 ligneAdresseIP
 AjoutArmoire, 18
 ligneChoixArmoire
 IHMPricipale, 113
 ligneConnexion
 IHMPricipale, 113
 ligneIndicateur
 IHMPricipale, 113
 ligneMotDePasse
 IHMPricipale, 113
 ligneNomArmoire
 AjoutArmoire, 18
 ligneNomUtilisateurs
 ModificationUtilisateur, 122
 SuppressionUtilisateur, 147
 ligneTitre
 AjoutArmoire, 18
 AjoutUtilisateur, 22
 GestionCasier, 71
 ModificationUtilisateur, 122
 MouvementsArmoire, 135
 SuppressionUtilisateur, 148
 UtilisateursArmoire, 174
 ligneUtilisateur
 IHMPricipale, 113
 lignem_badge
 AjoutUtilisateur, 22
 ModificationUtilisateur, 121
 SuppressionUtilisateur, 147
 lignem_dateDeValdite
 AjoutUtilisateur, 22
 ModificationUtilisateur, 121
 SuppressionUtilisateur, 147
 lignem_identifiant
 AjoutUtilisateur, 22
 ModificationUtilisateur, 121
 SuppressionUtilisateur, 147
 lignem_nom
 AjoutUtilisateur, 22
 ModificationUtilisateur, 121
 SuppressionUtilisateur, 147
 lignem_nomGroupe
 AjoutUtilisateur, 22
 ModificationUtilisateur, 122
 SuppressionUtilisateur, 147
 lignem_nomProfil
 AjoutUtilisateur, 22
 ModificationUtilisateur, 122
 SuppressionUtilisateur, 147
 lignem_prenom
 AjoutUtilisateur, 22
 ModificationUtilisateur, 122
 SuppressionUtilisateur, 147
 lire
 Armoires, 29
 Casiers, 63
 Mouvements, 132
 Utilisateur, 154
 UtilisateursArmoire, 170
 lireAdresseIP
 IhmGestionArmoire, 96
 lireBadge
 AuthentificationBadge, 36
 lireGroupe
 UtilisateursArmoire, 171
 lirePort
 GestionBalances, 68
 lireProfil
 UtilisateursArmoire, 171
 lireUtilisateur
 Utilisateur, 155
 UtilisateurTerminal, 179
 listeArmoires
 Armoires, 30
 listeArmoiresChanged
 Armoires, 29
 listeCasiers
 Casiers, 65
 listeCasiersChanged
 Casiers, 64
 listeChoixArmoire
 IHMPricipale, 113
 listeGroupe
 AjoutUtilisateur, 22
 ModificationUtilisateur, 122
 SuppressionUtilisateur, 148
 UtilisateursArmoire, 175
 listeGroupeChanged
 UtilisateursArmoire, 171
 listeMouvements
 Mouvements, 134
 listeMouvementsChanged
 Mouvements, 133
 listeNomUtilisateurs
 ModificationUtilisateur, 122
 SuppressionUtilisateur, 148
 listeProfil
 AjoutUtilisateur, 23
 ModificationUtilisateur, 122
 SuppressionUtilisateur, 148
 UtilisateursArmoire, 176
 listeProfilChanged
 UtilisateursArmoire, 172
 listeUtilisateursArmoire
 UtilisateursArmoire, 176
 listeUtilisateursArmoireChanged
 UtilisateursArmoire, 172
 listerGroupe
 IhmGestionGroupe, 106
 logo

- IHMPrincipale, 113
- logoProjet
 - IHMPrincipale, 114
- m_action
 - Mouvement, 129
- m_adresseIPArmoire
 - Casiers, 65
- m_adresseIP
 - Armoire, 26
- m_armoires
 - Armoires, 30
- m_badge
 - UtilisateurArmoire, 166
 - UtilisateurTerminal, 181
- m_casier
 - Mouvement, 129
- m_contenu
 - Mouvement, 129
- m_dateDeValidite
 - UtilisateurArmoire, 166
- m_dateValidite
 - UtilisateurTerminal, 181
- m_estAutorise
 - UtilisateurTerminal, 182
- m_estExistant
 - UtilisateurTerminal, 182
- m_groupe
 - UtilisateurTerminal, 182
- m_horodatage
 - Mouvement, 130
- m_id
 - Groupe, 74
 - Profil, 139
- m_idGroupe
 - UtilisateurTerminal, 182
- m_idProfil
 - UtilisateurTerminal, 182
- m_idUtilisateur
 - UtilisateurArmoire, 166
 - UtilisateurTerminal, 182
- m_identifiant
 - UtilisateurArmoire, 166
 - UtilisateurTerminal, 182
- m_listeGroupe
 - UtilisateursArmoire, 174
- m_listeProfil
 - UtilisateursArmoire, 174
- m_mouvements
 - Mouvements, 133
- m_nbCasiers
 - Casiers, 65
- m_nom
 - Armoire, 26
 - Groupe, 74
 - Profil, 139
 - UtilisateurArmoire, 166
 - UtilisateurTerminal, 183
- m_nomGroupe
 - UtilisateurArmoire, 166
- m_nomProfil
 - UtilisateurArmoire, 166
- m_pBaseDeDonnees
 - Armoires, 30
 - Casiers, 65
 - Mouvements, 133
 - UtilisateurTerminal, 183
 - UtilisateursArmoire, 174
- m_prenom
 - UtilisateurArmoire, 167
 - UtilisateurTerminal, 183
- m_profil
 - UtilisateurTerminal, 183
- m_quantite
 - Mouvement, 130
- m_timer
 - Casiers, 65
- m_utilisateur
 - Mouvement, 130
- m_utilisateursArmoire
 - UtilisateursArmoire, 175
- mail
 - Administrateur, 15
- main
 - ihm-estock/main.cpp, 197
 - terminalmobile/main.cpp, 198
- main.cpp, 197, 198
- MenuArmoire, 115
 - armoire, 118
 - button, 116
 - button1, 116
 - button4, 116
 - button5, 116
 - columnLayout, 117
 - height, 117
 - labelArmoire, 117
 - pageMouvementsArmoire, 117
 - pageStockArmoire, 117
 - pageUtilisateursArmoire, 117
 - title, 117
 - utilisateur, 118
 - width, 117
- MenuArmoire.qml, 199
- modal
 - AjoutArmoire, 18
 - AjoutUtilisateur, 23
 - GestionCasier, 71
 - ModificationUtilisateur, 122
 - SuppressionUtilisateur, 148
- ModificationUtilisateur, 118
 - boutonAnnuler, 120
 - boutonModifieur, 120
 - boutons, 120
 - edition, 120
 - footer, 120
 - labelNomUtilisateurs, 121
 - labelTitre, 121
 - labelm_badge, 120
 - labelm_dateDeValidite, 120
 - labelm_identifiant, 120
 - labelm_nom, 120
 - labelm_nomGroupe, 121

- labelm_nomProfil, 121
- labelm_prenom, 121
- ligneNomUtilisateurs, 122
- ligneTitre, 122
- lignem_badge, 121
- lignem_dateDeValidite, 121
- lignem_identifiant, 121
- lignem_nom, 121
- lignem_nomGroupe, 122
- lignem_nomProfil, 122
- lignem_prenom, 122
- listeGroupe, 122
- listeNomUtilisateurs, 122
- listeProfil, 122
- modal, 122
- onOpened, 123
- saisiem_badge, 123
- saisiem_dateDeValidite, 123
- saisiem_identifiant, 123
- saisiem_nom, 123
- saisiem_prenom, 123
- ModificationUtilisateur.qml, 200
- modifier
 - UtilisateursArmoire, 172
- modifierGroupe
 - IhmGestionGroupe, 107
- motDePasse
 - Administrateur, 15
 - IhmAuthenticationSansBadge, 89
 - Utilisateur, 158
- mousePressEvent
 - Casier, 56
- Mouvement, 124
 - actionChanged, 125
 - casierChanged, 125
 - contenuChanged, 125
 - getAction, 126
 - getCasier, 126
 - getContenu, 126
 - getHorodatage, 126
 - getQuantite, 126
 - getUtilisateur, 127
 - horodatageChanged, 127
 - m_action, 129
 - m_casier, 129
 - m_contenu, 129
 - m_horodatage, 130
 - m_quantite, 130
 - m_utilisateur, 130
 - Mouvement, 125
 - quantiteChanged, 127
 - setAction, 127
 - setCasier, 127
 - setContenu, 128
 - setHorodatage, 128
 - setMouvement, 128
 - setQuantite, 128
 - setUtilisateur, 129
 - utilisateurChanged, 129
- mouvement.cpp, 200
- mouvement.h, 200
- Mouvements, 130
 - ~Mouvements, 132
 - getMouvements, 132
 - lire, 132
 - listeMouvements, 134
 - listeMouvementsChanged, 133
 - m_mouvements, 133
 - m_pBaseDeDonnees, 133
 - Mouvements, 132
- mouvements.cpp, 200
- mouvements.h, 200
- MouvementsArmoire, 134
 - boutonRecharger, 135
 - boutonRetour, 135
 - height, 135
 - labelTitre, 135
 - ligneTitre, 135
 - onVisibleChanged, 136
 - row, 136
 - rowButton, 136
 - title, 136
 - vueMouvement, 136
 - width, 136
- MouvementsArmoire.qml, 201
- mutex
 - BaseDeDonnees, 51
- NB_BASEDEDONNEES
 - terminalmobile/baseDeDonnees.h, 188
- NB_CASIER_MAX
 - IhmGestionArmoire.h, 195
- nbAcces
 - BaseDeDonnees, 51
- nbCasiersChanged
 - Casiers, 64
- Nom
 - UtilisateurTerminal, 183
- nom
 - Administrateur, 15
 - Utilisateur, 158
- nomArticle
 - Article, 33
 - Casier, 60
- nomArticleChanged
 - Casier, 57
- nomBDD
 - BaseDeDonnees, 51
- nomCasier
 - Casier, 60
- nomChanged
 - Armoire, 25
 - Groupe, 73
 - Profil, 138
 - UtilisateurArmoire, 163
- nomGroupeChanged
 - UtilisateurArmoire, 163
- nomProfilChanged
 - UtilisateurArmoire, 163
- numero
 - Casier, 59
- numeroCasier

- Casier, [60](#)
- numeroCasierChanged
 - Casier, [57](#)
- onOpened
 - AjoutUtilisateur, [23](#)
 - ModificationUtilisateur, [123](#)
 - SuppressionUtilisateur, [148](#)
- onVisibleChanged
 - MouvementsArmoire, [136](#)
 - StockArmoire, [142](#)
 - UtilisateursArmoire, [175](#)
- ouvert
 - Casier, [59](#)
- ouvrir
 - BaseDeDonnees, [45, 46](#)
 - Casier, [57](#)
- ouvrirPort
 - GestionBalances, [68](#)
- PORT
 - GestionBalances.h, [192](#)
- pageMenuArmoire
 - IHMPrincipale, [114](#)
- pageMouvementsArmoire
 - MenuArmoire, [117](#)
- pageStockArmoire
 - MenuArmoire, [117](#)
- pageUtilisateursArmoire
 - MenuArmoire, [117](#)
- parametrerPort
 - GestionBalances, [68](#)
- passerSansBadge
 - IhmAuthentication, [79](#)
- port
 - GestionBalances, [69](#)
- Prenom
 - UtilisateurTerminal, [183](#)
- prenom
 - Administrateur, [15](#)
 - Utilisateur, [158](#)
- prenomChanged
 - UtilisateurArmoire, [163](#)
- Profil, [137](#)
 - getId, [138](#)
 - getNom, [138](#)
 - idChanged, [138](#)
 - m_id, [139](#)
 - m_nom, [139](#)
 - nomChanged, [138](#)
 - Profil, [137](#)
 - setId, [138](#)
 - setNom, [139](#)
- profil
 - Utilisateur, [158](#)
- profil.cpp, [201](#)
- profil.h, [201](#)
- quantite
 - Article, [33](#)
 - Casier, [60](#)
- quantiteChanged
 - Casier, [57](#)
 - Mouvement, [127](#)
- README.md, [201](#)
- radius
 - Casier, [59](#)
- receptionDonneeUtilisateur
 - Utilisateur, [156](#)
- rechercherArticle
 - Article, [32](#)
 - IhmGestionArmoire, [96](#)
- recuperer
 - BaseDeDonnees, [46–50](#)
- recupererArticle
 - Article, [32](#)
- reinitialiser
 - UtilisateurTerminal, [181](#)
- reinitialiserSaisieBadge
 - IhmAuthentication, [79](#)
- restituerArticle
 - Article, [32](#)
- row
 - MouvementsArmoire, [136](#)
 - UtilisateursArmoire, [175](#)
- rowButton
 - MouvementsArmoire, [136](#)
 - UtilisateursArmoire, [175](#)
- saisieAdresseIP
 - AjoutArmoire, [18](#)
- saisieMotDePasse
 - IHMPrincipale, [114](#)
- saisieNomArmoire
 - AjoutArmoire, [18](#)
- saisieUtilisateur
 - IHMPrincipale, [114](#)
- saisiem_badge
 - AjoutUtilisateur, [23](#)
 - ModificationUtilisateur, [123](#)
 - SuppressionUtilisateur, [148](#)
- saisiem_dateDeValidite
 - AjoutUtilisateur, [23](#)
 - ModificationUtilisateur, [123](#)
 - SuppressionUtilisateur, [148](#)
- saisiem_identifiant
 - AjoutUtilisateur, [23](#)
 - ModificationUtilisateur, [123](#)
 - SuppressionUtilisateur, [148](#)
- saisiem_nom
 - AjoutUtilisateur, [23](#)
 - ModificationUtilisateur, [123](#)
 - SuppressionUtilisateur, [149](#)
- saisiem_prenom
 - AjoutUtilisateur, [23](#)
 - ModificationUtilisateur, [123](#)
 - SuppressionUtilisateur, [149](#)
- seConnecter
 - GestionBalances, [69](#)
- seDeconnecter
 - GestionBalances, [69](#)

selectionnerArticle
 IhmGestionArmoire, 97
 selectionnerCasier
 IhmGestionArmoire, 97
 selectionnerGroupe
 IhmGestionGroupe, 107
 setAction
 Mouvement, 127
 setAdresseIP
 Armoire, 25
 setArmoire
 Armoire, 26
 setBadge
 UtilisateurArmoire, 163
 setCasier
 Casier, 57
 Mouvement, 127
 setContenu
 Mouvement, 128
 setDateDeValidite
 UtilisateurArmoire, 163
 setDisponible
 Casier, 58
 setHorodatage
 Mouvement, 128
 setId
 Groupe, 73
 Profil, 138
 setIdUtilisateur
 UtilisateurArmoire, 164
 setIdentifiant
 UtilisateurArmoire, 164
 setMouvement
 Mouvement, 128
 setNom
 Administrateur, 14
 Armoire, 26
 Groupe, 74
 Profil, 139
 Utilisateur, 156
 UtilisateurArmoire, 164
 setNomArticle
 Casier, 58
 setNomGroupe
 UtilisateurArmoire, 164
 setNomProfil
 UtilisateurArmoire, 165
 setNumeroCasier
 Casier, 58
 setPrenom
 Administrateur, 14
 Utilisateur, 156
 UtilisateurArmoire, 165
 setQuantite
 Casier, 58
 Mouvement, 128
 setUtilisateur
 IhmGestionArmoire, 98
 IhmGestionGroupe, 108
 Mouvement, 129
 UtilisateurArmoire, 165
 showEvent
 IhmAuthenticationSansBadge, 87
 IhmGestionArmoire, 98
 IhmGestionGroupe, 108
 StockArmoire, 140
 afficherCasiers, 141
 armoire, 143
 boutonQuitterStockArmoire, 141
 colonneBoutons, 141
 colonneCasier1, 142
 colonneCasier2, 142
 colonneCasiersDroite, 142
 colonneCasiersGauche, 142
 conteneurCasiers, 142
 dialogueGestionCasier, 142
 height, 142
 labelUtilisateur, 142
 onVisibleChanged, 142
 title, 143
 utilisateur, 143
 width, 143
 StockArmoire.qml, 201
 SuppressionUtilisateur, 143
 boutonAnnuler, 145
 boutonSupprimer, 145
 boutons, 145
 edition, 145
 footer, 146
 labelNomUtilisateurs, 146
 labelTitre, 147
 labelm_badge, 146
 labelm_dateDeValidite, 146
 labelm_identifiant, 146
 labelm_nom, 146
 labelm_nomGroupe, 146
 labelm_nomProfil, 146
 labelm_prenom, 146
 ligneNomUtilisateurs, 147
 ligneTitre, 148
 lignem_badge, 147
 lignem_dateDeValidite, 147
 lignem_identifiant, 147
 lignem_nom, 147
 lignem_nomGroupe, 147
 lignem_nomProfil, 147
 lignem_prenom, 147
 listeGroupe, 148
 listeNomUtilisateurs, 148
 listeProfil, 148
 modal, 148
 onOpened, 148
 saisiem_badge, 148
 saisiem_dateDeValidite, 148
 saisiem_identifiant, 148
 saisiem_nom, 149
 saisiem_prenom, 149
 SuppressionUtilisateur.qml, 202
 supprimer
 Armoires, 30
 UtilisateursArmoire, 172

- supprimerClavierVirtuel
 - IhmAuthenticationSansBadge, 87
 - IhmGestionArmoire, 99
- supprimerGroupe
 - IhmGestionGroupe, 108
- terminalmobile/baseDeDonnees.h
 - BDD_MYSQL, 188
 - BDD_SQLITE, 188
 - NB_BASEDEDONNEES, 188
- terminalmobile/main.cpp
 - main, 198
- title
 - IHMPrincipale, 114
 - MenuArmoire, 117
 - MouvementsArmoire, 136
 - StockArmoire, 143
 - UtilisateursArmoire, 175
- typeBase
 - BaseDeDonnees, 51
- Ui, 12
- ui
 - IhmAuthentication, 81
 - IhmAuthenticationSansBadge, 89
 - IhmGestionArmoire, 100
 - IhmGestionBalances, 101
 - IhmGestionGroupe, 109
- Utilisateur, 149
 - ~Utilisateur, 152
 - badge, 157
 - bdd, 157
 - dateValidite, 157
 - email, 157
 - estAdministrateur, 152
 - estGestionnaire, 152
 - estUtilisateur, 153
 - getGroupe, 153
 - getIdGroupe, 153
 - getIdUtilisateur, 153
 - getNom, 154
 - getPrenom, 154
 - getProfil, 154
 - groupe, 157
 - idGroupe, 158
 - idProfil, 158
 - idUtilisateur, 158
 - identifiant, 157
 - lire, 154
 - lireUtilisateur, 155
 - motDePasse, 158
 - nom, 158
 - prenom, 158
 - profil, 158
 - receptionDonneeUtilisateur, 156
 - setNom, 156
 - setPrenom, 156
 - Utilisateur, 151
 - valide, 159
- utilisateur
 - IhmAuthentication, 81
 - IhmGestionArmoire, 100
 - IhmGestionGroupe, 109
 - MenuArmoire, 118
 - StockArmoire, 143
- utilisateur.cpp, 202
- utilisateur.h, 202
- UtilisateurArmoire, 159
 - badgeChanged, 161
 - dateDeValiditeChanged, 161
 - getBadge, 161
 - getDateDeValidite, 161
 - getIdUtilisateur, 161
 - getIdentifiant, 161
 - getNom, 162
 - getNomGroupe, 162
 - getNomProfil, 162
 - getPrenom, 162
 - idUtilisateurChanged, 163
 - identifiantChanged, 162
 - m_badge, 166
 - m_dateDeValidite, 166
 - m_idUtilisateur, 166
 - m_identifiant, 166
 - m_nom, 166
 - m_nomGroupe, 166
 - m_nomProfil, 166
 - m_prenom, 167
 - nomChanged, 163
 - nomGroupeChanged, 163
 - nomProfilChanged, 163
 - prenomChanged, 163
 - setBadge, 163
 - setDateDeValidite, 163
 - setIdUtilisateur, 164
 - setIdentifiant, 164
 - setNom, 164
 - setNomGroupe, 164
 - setNomProfil, 165
 - setPrenom, 165
 - setUtilisateurArmoire, 165
 - UtilisateurArmoire, 160
- utilisateurArmoire.cpp, 202
- utilisateurArmoire.h, 203
- utilisateurAutorise
 - IhmAuthenticationSansBadge, 87
- utilisateurChanged
 - Mouvement, 129
- UtilisateurTerminal, 176
 - ~UtilisateurTerminal, 178
 - estAutorise, 178
 - estExistant, 179
 - getNom, 179
 - getPrenom, 179
 - lireUtilisateur, 179
 - m_badge, 181
 - m_dateValidite, 181
 - m_estAutorise, 182
 - m_estExistant, 182
 - m_groupe, 182
 - m_idGroupe, 182
 - m_idProfil, 182

- m_idUtilisateur, 182
- m_identifiant, 182
- m_nom, 183
- m_pBaseDeDonnees, 183
- m_prenom, 183
- m_profil, 183
- Nom, 183
- Prenom, 183
- reinitialiser, 181
- UtilisateurTerminal, 178
- utilisateurUpdated, 181
- utilisateurTerminal.cpp, 204
- utilisateurTerminal.h, 204
- utilisateurUpdated
 - UtilisateurTerminal, 181
- utilisateurs
 - IhmGestionGroupe, 110
- UtilisateursArmoire, 167
 - ~UtilisateursArmoire, 169
 - boutonAjouter, 173
 - boutonModifier, 173
 - boutonRecharger, 173
 - boutonRetour, 173
 - boutonSupprimer, 173
 - dialogueAjoutUtilisateur, 174
 - dialogueModificationUtilisateur, 174
 - dialogueSuppressionUtilisateur, 174
 - getListeGroupe, 169
 - getListeProfil, 169
 - getUtilisateursArmoire, 169
 - height, 174
 - inserer, 170
 - labelTitre, 174
 - ligneTitre, 174
 - lire, 170
 - lireGroupe, 171
 - lireProfil, 171
 - listeGroupe, 175
 - listeGroupeChanged, 171
 - listeProfil, 176
 - listeProfilChanged, 172
 - listeUtilisateursArmoire, 176
 - listeUtilisateursArmoireChanged, 172
 - m_listeGroupe, 174
 - m_listeProfil, 174
 - m_pBaseDeDonnees, 174
 - m_utilisateursArmoire, 175
 - modifier, 172
 - onVisibleChanged, 175
 - row, 175
 - rowButton, 175
 - supprimer, 172
 - title, 175
 - UtilisateursArmoire, 169
 - vueUtilisateurArmoire, 175
 - width, 175
- utilisateursArmoire.cpp, 203
- utilisateursArmoire.h, 203
- UtilisateursArmoire.qml, 203

- valide
 - AuthenticationBadge, 38
 - IhmAuthenticationSansBadge, 89
 - Utilisateur, 159
- validerAcces
 - IhmAuthentication, 79
 - IhmAuthenticationSansBadge, 87
- visible
 - Casier, 59
 - IHMPrincipale, 114
- visualiserLesMouvements
 - Article, 32
- vueMouvement
 - MouvementsArmoire, 136
- vueUtilisateurArmoire
 - UtilisateursArmoire, 175

- width
 - Casier, 60
 - IHMPrincipale, 114
 - MenuArmoire, 117
 - MouvementsArmoire, 136
 - StockArmoire, 143
 - UtilisateursArmoire, 175