

# GrOOm

version 0.2

BTS SNIR LaSalle Avignon 2020

## Table des matières

### 1 Le projet

Le portier connecté « grOOm » permettra à l'occupant du bureau de communiquer sa disponibilité avec des personnes extérieures (visiteurs). Pour cela, il utilise une application soit en version PC soit mobile.

#### 1.1 Table des matières

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

#### 1.2 Informations

##### Auteur

Etienne Valette [valette.etienne@gmail.com](mailto:valette.etienne@gmail.com)

##### Date

2020

##### Version

0.2

##### Voir également

<https://svn.riouxsvn.com/groom/>

### 2 Changelog

r1 | www-data | 2020-02-01 15 :03 :29 +0100 (sam. 01 févr. 2020) | 1 ligne

Creating initial repository structure

### 3 README

#### 3.1 Projet

##### 3.1.1 Présentation

Le portier connecté « grOOm » permettra à l'occupant du bureau de communiquer sa disponibilité avec des personnes extérieures (visiteurs). Pour cela, il utilise une application soit en version PC soit mobile.

Tout en s'intégrant facilement à l'environnement, il résout le manque d'interface entre les utilisateurs et les bureaux permettant de travailler plus efficacement.

A partir de l'application, l'occupant informe le visiteur de son état : "Libre", "Occupé" ou "Absent". Il aura la possibilité d'ajouter un message "libre" qui s'affichera alors sur l'écran du portier. S'il le souhaite, il informera le visiteur que celui-ci peut "Entrer" (ou "Entrez"). L'occupant dispose d'un mode "Sonnette" qu'il peut activer sur le portier connecté. Dans ce mode, le visiteur pourra sonner via l'écran tactile et prévenir l'occupant de sa présence.

Version : PC Desktop Qt

### 3.1.2 Informations

#### Auteur

Etienne Valette [valette.etienne@gmail.com](mailto:valette.etienne@gmail.com)

#### Date

2020

#### Version

0.2

#### Voir également

<https://svn.riouxsvn.com/groom/>

## 4 A propos

#### Auteur

Etienne Valette [valette.etienne@gmail.com](mailto:valette.etienne@gmail.com)

#### Date

2020

#### Version

0.2

#### Voir également

<https://svn.riouxsvn.com/groom/>

## 5 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

## 6 Liste des choses à faire

**Membre `IhmGroom` :** `IhmGroom (QWidget *parent=nullptr)`

Renommer les widgets

## 7 Documentation des espaces de nommage

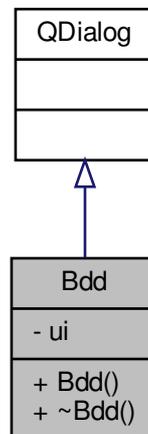
### 7.1 Référence de l'espace de nommage Ui

## 8 Documentation des classes

### 8.1 Référence de la classe Bdd

```
#include <bdd.h>
```

Graphe de collaboration de Bdd :



#### Fonctions membres publiques

- `Bdd(QWidget *parent=nullptr)`
- `~Bdd()`

#### Attributs privés

- `Ui : :Bdd * ui`

#### 8.1.1 Description détaillée

Définition à la ligne 10 du fichier `bdd.h`.

#### 8.1.2 Documentation des constructeurs et destructeur

### 8.1.2.1 Bdd()

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

Définition à la ligne 4 du fichier [bdd.cpp](#).

Références [ui](#).

```
00004         :
00005     QDialog(parent),
00006     ui(new Ui::Bdd)
00007 {
00008     ui->setupUi(this);
00009 }
```

### 8.1.2.2 ~Bdd()

```
Bdd::~Bdd ( )
```

Définition à la ligne 11 du fichier [bdd.cpp](#).

Références [ui](#).

```
00012 {
00013     delete ui;
00014 }
```

## 8.1.3 Documentation des données membres

### 8.1.3.1 ui

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

Définition à la ligne 19 du fichier [bdd.h](#).

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

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

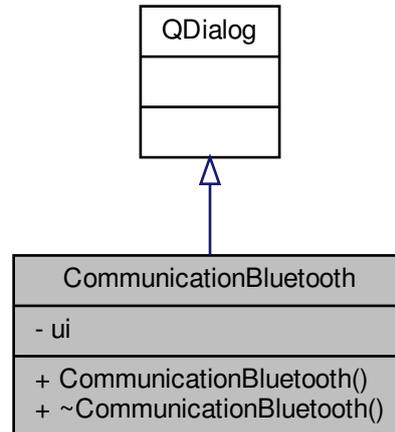
- [bdd.h](#)
- [bdd.cpp](#)

## 8.2 Référence de la classe CommunicationBluetooth

Class permettant de mettre en place une communication bluetooth.

```
#include "communicationbluetooth.h"
```

Graphe de collaboration de CommunicationBluetooth :



### Fonctions membres publiques

- [CommunicationBluetooth](#) ([QWidget](#) \*parent=nullptr)  
*Constructeur classe [CommunicationBluetooth](#).*
- [~CommunicationBluetooth](#) ()  
*Destructeur classe [CommunicationBluetooth](#).*

### Attributs privés

- [Ui](#) : :CommunicationBluetooth \* [ui](#)

#### 8.2.1 Description détaillée

Class permettant de mettre en place une communication bluetooth.

Définition à la ligne 17 du fichier [communicationbluetooth.h](#).

#### 8.2.2 Documentation des constructeurs et destructeur

##### 8.2.2.1 CommunicationBluetooth()

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

Constructeur classe [CommunicationBluetooth](#).

## Paramètres

<i>parent</i>	
---------------	--

Définition à la ligne 4 du fichier [communicationbluetooth.cpp](#).

Références [ui](#).

```
00004                                     :
00005         QDialog(parent),
00006         ui(new Ui::CommunicationBluetooth)
00007 {
00008     qDebug() << Q_FUNC_INFO;
00009     ui->setupUi(this);
00010 }
```

## 8.2.2.2 ~CommunicationBluetooth()

```
CommunicationBluetooth::~CommunicationBluetooth ( )
```

Destructeur classe [CommunicationBluetooth](#).

## Paramètres

<i>parent</i>	
---------------	--

Définition à la ligne 12 du fichier [communicationbluetooth.cpp](#).

Références [ui](#).

```
00013 {
00014     delete ui;
00015     qDebug() << Q_FUNC_INFO;
00016 }
```

## 8.2.3 Documentation des données membres

## 8.2.3.1 ui

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

Définition à la ligne 38 du fichier [communicationbluetooth.h](#).

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

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

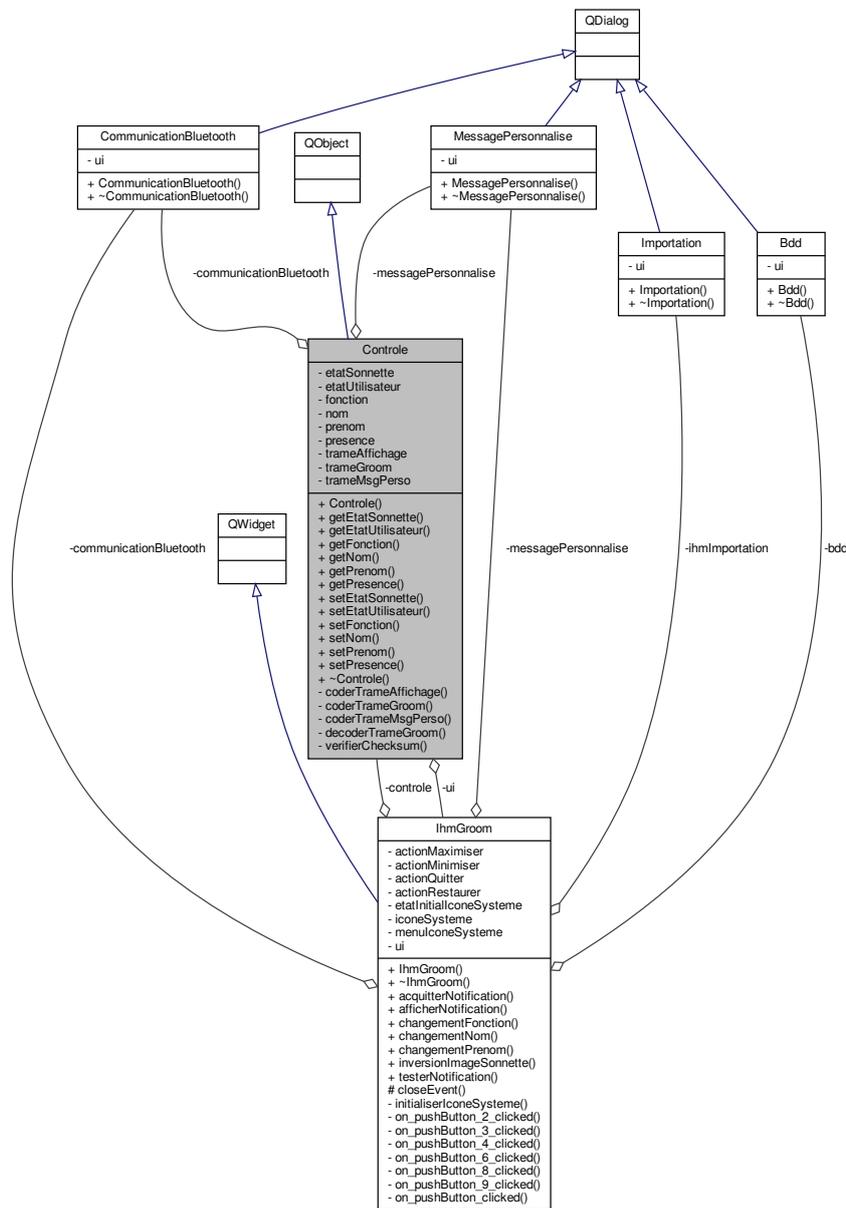
- [communicationbluetooth.h](#)
- [communicationbluetooth.cpp](#)

## 8.3 Référence de la classe Controle

Classe assurant le controle et la validé de l'envoi et réception des trames.

```
#include "controle.h"
```

Grphe de collaboration de Controle :



## Fonctions membres publiques

- **Controle** (**QObject** \*parent=nullptr)  
Constructeur classe *Controle*.
- bool **getEtatSonnette** ()  
retourne l'état de la sonnette
- int **getEtatUtilisateur** ()  
retourne l'état de la sonnette
- QString **getFonction** ()

- *retroune la fonction*  
— QString [getNom](#) ()
- *retroune le nom*  
— QString [getPrenom](#) ()
- *retroune le prenom*  
— bool [getPresence](#) ()
- *retroune l'état de la sonnette*  
— void [setEtatSonnette](#) (bool newEtatSonnette)
- *modifie l'état de la sonnette*  
— void [setEtatUtilisateur](#) (int newEtatUtilisateur)
- *modifie l'état de la sonnette*  
— void [setFonction](#) (QString newFonction)
- *modifie la fonction*  
— void [setNom](#) (QString newNom)
- *modifie le nom*  
— void [setPrenom](#) (QString newPrenom)
- void [setPresence](#) (bool newPresence)
- *modifie l'état de la présence*  
— [~Controle](#) ()
- *Destructeur classe [Controle](#).*

### Fonctions membres privées

- void [coderTrameAffichage](#) (QString [trameAffichage](#))  
*coder la trame Affichage*
- void [coderTrameGroom](#) (QString [trameGroom](#))  
*coder la trame Groom*
- void [coderTrameMsgPerso](#) (QString [trameMsgPerso](#))  
*coder la trame Msg Perso*
- void [decoderTrameGroom](#) (QString [trameGroom](#))  
*Décode la trame Groom.*
- bool [verifierChecksum](#) (QString [trameGroom](#))  
*Regarde le checksum de la trameGroom.*

### Attributs privés

- [CommunicationBluetooth](#) \* [communicationBluetooth](#)
- bool [etatSonnette](#)
- int [etatUtilisateur](#)
- QString [fonction](#)
- [MessagePersonnalise](#) \* [messagePersonnalise](#)
- QString [nom](#)
- QString [prenom](#)
- bool [presence](#)
- QString [trameAffichage](#)
- QString [trameGroom](#)
- QString [trameMsgPerso](#)
- [IhmGroom](#) \* [ui](#)

#### 8.3.1 Description détaillée

Classe assurant le controle et la validé de l'envoi et réception des trames.

Définition à la ligne 22 du fichier [controle.h](#).

#### 8.3.2 Documentation des constructeurs et destructeur

##### 8.3.2.1 Controle()

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

Constructeur classe [Controle](#).

Attribut de la fonction de l'utilisateur

## Paramètres

<i>parent</i>
---------------

Définition à la ligne 5 du fichier [controle.cpp](#).

Références [etatSonnette](#).

```
00005                                     : QObject (parent)
00006 {
00007     qDebug() << Q_FUNC_INFO;
00008     //     ui = new IhmGroom;
00009     etatSonnette = true;
00010
00011 }
```

## 8.3.2.2 ~Controle()

```
Controle::~Controle ( )
```

Destructeur classe [Controle](#).

## Paramètres

<i>parent</i>
---------------

Définition à la ligne 13 du fichier [controle.cpp](#).

```
00014 {
00015     qDebug() << Q_FUNC_INFO;
00016 }
```

## 8.3.3 Documentation des fonctions membres

## 8.3.3.1 coderTrameAffichage()

```
Controle::coderTrameAffichage (
    QString trameAffichage ) [private]
```

coder la trame Affichage

Définition à la ligne 101 du fichier [controle.cpp](#).

Références [IhmGroom : :changementFonction\(\)](#), [IhmGroom : :changementNom\(\)](#), [IhmGroom : :changementPrenom\(\)](#), [setFonction\(\)](#), [setNom\(\)](#), [setPrenom\(\)](#), et [ui](#).

```
00102 {
00103     setNom(trameAffichage.section(";",1,1));
00104
00105     setPrenom(trameAffichage.section(";",2,2));
00106
00107     setFonction(trameAffichage.section(";",3,3));
00108
00109     ui->changementNom();
00110     ui->changementPrenom();
00111     ui->changementFonction();
00112 }
```

### 8.3.3.2 coderTrameGroom()

```
Controle::coderTrameGroom (
    QString trameGroom ) [private]
```

coder la trame Groom

### 8.3.3.3 coderTrameMsgPerso()

```
Controle::coderTrameMsgPerso (
    QString trameMsgPerso ) [private]
```

coder la trame Msg Perso

### 8.3.3.4 decoderTrameGroom()

```
Controle::decoderTrameGroom (
    QString trameGroom ) [private]
```

Décode la trame Groom.

Définition à la ligne 49 du fichier [controle.cpp](#).

Références [setEtatSonnette\(\)](#), [setEtatUtilisateur\(\)](#), et [setPresence\(\)](#).

```
00050 {
00051
00052     setEtatUtilisateur((trameGroom.section(";",1,1)).toInt());
00053
00054
00055     if (trameGroom.section(";",2,2) == 1)
00056     {
00057         setEtatSonnette(true);
00058     }
00059     else
00060     {
00061         setEtatSonnette(false);
00062     }
00063
00064     if (trameGroom.section(";",3,3) == 1)
00065     {
00066         setPresence(true);
00067     }
00068     else
00069     {
00070         setPresence(false);
00071     }
00072
00073 }
```

### 8.3.3.5 getEtatSonnette()

```
Controle::getEtatSonnette ( )
```

retrouve l'état de la sonnette

Renvoie

etatSonnette

Définition à la ligne 33 du fichier [controle.cpp](#).

Références [etatSonnette](#).

Référencé par [IhmGroom : :inversionImageSonnette\(\)](#).

```
00034 {
00035     return etatSonnette;
00036 }
```

### 8.3.3.6 getEtatUtilisateur()

Controle::getEtatUtilisateur ( )

retrouve l'état de la sonnette

**Renvoie**

etatUtilisateur

Définition à la ligne 23 du fichier [controle.cpp](#).

Références [etatUtilisateur](#).

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

### 8.3.3.7 getFonction()

Controle::getFonction ( )

retrouve la fonction

**Renvoie**

fonction

Définition à la ligne 96 du fichier [controle.cpp](#).

Références [fonction](#).

Référencé par [IhmGroom](#) : [:changementFonction\(\)](#).

```
00097 {  
00098     return fonction;  
00099 }
```

### 8.3.3.8 getNom()

Controle::getNom ( )

retrouve le nom

**Renvoie**

nom

Définition à la ligne 80 du fichier [controle.cpp](#).

Références [nom](#).

Référencé par [IhmGroom](#) : [:changementNom\(\)](#).

```
00081 {  
00082     return nom;  
00083 }
```

### 8.3.3.9 `getPrenom()`

`Controle::getPrenom ( )`

retrouve le prenom

#### Renvoie

prenom

Définition à la ligne 88 du fichier `controle.cpp`.

Références [prenom](#).

Référencé par `lhmGroom : :changementPrenom()`.

```
00089 {  
00090     return prenom;  
00091 }
```

### 8.3.3.10 `getPresence()`

`Controle::getPresence ( )`

retrouve l'état de la sonnette

#### Renvoie

presence

Définition à la ligne 43 du fichier `controle.cpp`.

Références [presence](#).

```
00044 {  
00045     return presence;  
00046 }
```

### 8.3.3.11 `setEtatSonnette()`

`Controle::setEtatSonnette (`  
    `bool newEtatSonnette )`

modifie l'état de la sonnette

#### Paramètres

<code>etatSonnette</code>	
---------------------------	--

Définition à la ligne 28 du fichier `controle.cpp`.

Références [etatSonnette](#).

Référencé par [decoderTrameGroom\(\)](#), et [IhmGroom : :inversionImageSonnette\(\)](#).

```
00029 {  
00030     etatSonnette = newEtatSonnette;  
00031 }
```

#### 8.3.3.12 setEtatUtilisateur()

```
Controle::setEtatUtilisateur (  
    int newEtatUtilisateur )
```

modifie l'état de la sonnette

##### Paramètres

<i>etatUtilisateur</i>	
------------------------	--

Définition à la ligne 18 du fichier [controle.cpp](#).

Références [etatUtilisateur](#).

Référencé par [decoderTrameGroom\(\)](#), [IhmGroom : :on\\_pushButton\\_2\\_clicked\(\)](#), [IhmGroom : :on\\_pushButton\\_3\\_clicked\(\)](#), et [IhmGroom : :on\\_pushButton\\_4\\_clicked\(\)](#).

```
00019 {  
00020     etatUtilisateur = newEtatUtilisateur;  
00021 }
```

#### 8.3.3.13 setFonction()

```
Controle::setFonction (  
    QString newFonction )
```

modifie la fonction

##### Paramètres

<i>fonction</i>	
-----------------	--

Définition à la ligne 92 du fichier [controle.cpp](#).

Références [fonction](#).

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

```
00093 {  
00094     fonction = newFonction;  
00095 }
```

#### 8.3.3.14 setNom()

```
Controle::setNom (
    QString newNom )
```

modifie le nom

##### Paramètres

<i>nom</i>	
------------	--

Définition à la ligne 75 du fichier [controle.cpp](#).

Références [nom](#).

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

```
00076 {
00077     nom = newNom;
00078 }
```

#### 8.3.3.15 setPrenom()

```
void Controle::setPrenom (
    QString newPrenom )
```

Définition à la ligne 84 du fichier [controle.cpp](#).

Références [prenom](#).

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

```
00085 {
00086     prenom = newPrenom;
00087 }
```

#### 8.3.3.16 setPresence()

```
Controle::setPresence (
    bool newPresence )
```

modifie l'état de la présence

##### Paramètres

<i>presence</i>	
-----------------	--

Définition à la ligne 38 du fichier [controle.cpp](#).

Références [presence](#).

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

```
00039 {  
00040     presence = newPresence;  
00041 }
```

#### 8.3.3.17 verifierChecksum()

```
Controle::verifierChecksum (  
    QString trameGroom ) [private]
```

Regarde le checksum de la trameGroom.

Trame à envoyer à la liaison série

##### Renvoie

retourne un true ou false

### 8.3.4 Documentation des données membres

#### 8.3.4.1 communicationBluetooth

```
CommunicationBluetooth* Controle::communicationBluetooth [private]
```

Définition à la ligne 27 du fichier [controle.h](#).

#### 8.3.4.2 etatSonnette

```
bool Controle::etatSonnette [private]
```

Attribut de l'état de l'utilisateur

Définition à la ligne 70 du fichier [controle.h](#).

Référencé par [Controle\(\)](#), [getEtatSonnette\(\)](#), et [setEtatSonnette\(\)](#).

#### 8.3.4.3 etatUtilisateur

```
int Controle::etatUtilisateur [private]
```

Définition à la ligne 69 du fichier [controle.h](#).

Référencé par [getEtatUtilisateur\(\)](#), et [setEtatUtilisateur\(\)](#).

#### 8.3.4.4 fonction

`QString Controle::fonction [private]`

Attribut du prénom de l'utilisateur

Définition à la ligne 75 du fichier [controle.h](#).

Référencé par [getFonction\(\)](#), et [setFonction\(\)](#).

#### 8.3.4.5 messagePersonnalise

`MessagePersonnalise* Controle::messagePersonnalise [private]`

Objet servant à récupérer les trame du Bluetooth

Définition à la ligne 28 du fichier [controle.h](#).

#### 8.3.4.6 nom

`QString Controle::nom [private]`

Attribut de l'état du capteur de présence

Définition à la ligne 73 du fichier [controle.h](#).

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

#### 8.3.4.7 prenom

`QString Controle::prenom [private]`

Attribut du nom de l'utilisateur

Définition à la ligne 74 du fichier [controle.h](#).

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

#### 8.3.4.8 presence

`bool Controle::presence [private]`

Attribut de l'état de la sonnette

Définition à la ligne 71 du fichier [controle.h](#).

Référencé par [getPresence\(\)](#), et [setPresence\(\)](#).

#### 8.3.4.9 trameAffichage

```
QString Controle::trameAffichage [private]
```

Trame provenant de la liaison Bluetooth et à envoyer

Définition à la ligne [32](#) du fichier [controle.h](#).

#### 8.3.4.10 trameGroom

```
QString Controle::trameGroom [private]
```

L'interface utilisateur

Définition à la ligne [31](#) du fichier [controle.h](#).

#### 8.3.4.11 trameMsgPerso

```
QString Controle::trameMsgPerso [private]
```

Trame à envoyer à la liaison série

Définition à la ligne [33](#) du fichier [controle.h](#).

#### 8.3.4.12 ui

```
IhmGroom* Controle::ui [private]
```

Objet servant à récupérer les messages personnalisé

Définition à la ligne [29](#) du fichier [controle.h](#).

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

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

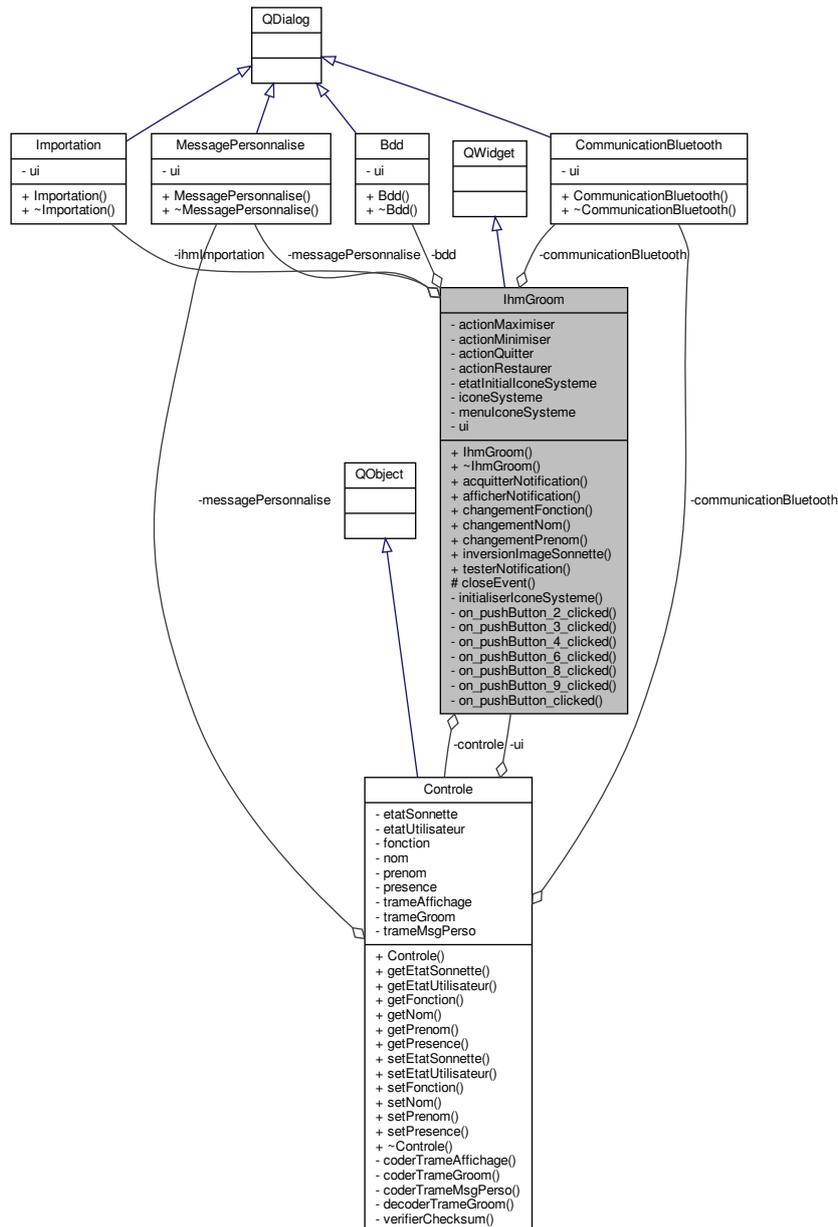
- [controle.h](#)
- [controle.cpp](#)

## 8.4 Référence de la classe IhmGroom

Déclaration de la classe [IhmGroom](#).

```
#include <ihmgroom.h>
```

Graphe de collaboration de IhmGroom :



### Connecteurs publics

- void [acquitterNotification](#) ()  
*Méthode qui permet d'acquitter une notification.*
- void [afficherNotification](#) (QString titre, QString message, int duree=1000)  
*Méthode qui permet d'afficher une notification système.*
- void [changementFonction](#) ()  
*change la fonction sur l'ihm*

- void `changementNom` ()  
*change le nom sur l'ihm*
- void `changementPrenom` ()  
*change le prenom sur l'ihm*
- void `inversionImageSonnette` ()  
*Inverse l'image de la sonnette.*
- void `testerNotification` ()

#### Fonctions membres publiques

- `IhmGroom` (QWidget \*parent=nullptr)  
*Constructeur de la classe IhmGroom.*
- `~IhmGroom` ()  
*Destructeur de la classe IhmGroom.*

#### Fonctions membres protégées

- void `closeEvent` (QCloseEvent \*event)  
*Méthode redéfinie qui est appelée automatiquement lors d'une demande de fermeture.*

#### Connecteurs privés

- void `on_pushButton_2_clicked` ()
- void `on_pushButton_3_clicked` ()  
*change l'état de la personne en Libre*
- void `on_pushButton_4_clicked` ()
- void `on_pushButton_6_clicked` ()  
*affiche l'ihm communication bluetooth*
- void `on_pushButton_8_clicked` ()  
*affiche l'ihm importation*
- void `on_pushButton_9_clicked` ()
- void `on_pushButton_clicked` ()  
*affiche l'ihm Message personnalisé*

#### Fonctions membres privées

- void `initialiserIcôneSysteme` ()  
*Méthode qui permet à l'application de s'installer dans la barre système.*

#### Attributs privés

- QAction \* `actionMaximiser`  
*L'action maximiser l'application.*
- QAction \* `actionMinimiser`  
*L'action minimiser l'application.*
- QAction \* `actionQuitter`  
*L'action quitter l'application.*
- QAction \* `actionRestaurer`  
*L'action restaurer l'application.*
- Bdd \* `bdd`
- CommunicationBluetooth \* `communicationBluetooth`  
*L'interface de la Configuration.*
- Controle \* `controle`
- bool `etatInitialIcôneSysteme`  
*Booléen indiquant si c'est la première demande Quitter.*
- QSystemTrayIcon \* `iconeSysteme`  
*L'icône de l'application pour la barre système.*
- Importation \* `ihmImportation`  
*L'interface de l'Importation.*
- QMenu \* `menuIcôneSysteme`  
*Le menu de l'application.*
- MessagePersonnalise \* `messagePersonnalise`  
*L'interface du Message Personnalisé*
- Ui : :IhmGroom \* `ui`  
*L'interface utilisateur.*

### 8.4.1 Description détaillée

Déclaration de la classe [IhmGroom](#).

Définition à la ligne 54 du fichier [ihmgroom.h](#).

### 8.4.2 Documentation des constructeurs et destructeur

#### 8.4.2.1 IhmGroom()

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

Constructeur de la classe [IhmGroom](#).

Constructeur classe [IhmGroom](#).

#### Paramètres

<i>parent</i>	L'objet parent Qt (0 = fenêtre principale)
<i>parent</i>	

#### A faire Renommer les widgets

Définition à la ligne 31 du fichier [ihmgroom.cpp](#).

Références [bdd](#), [communicationBluetooth](#), [controle](#), [ihmImportation](#), [initialiserIcôneSysteme\(\)](#), [inversionImageSonnette\(\)](#), [message←→Personnalise](#), [NOM\\_APP](#), [testerNotification\(\)](#), et [ui](#).

```
00031                                     : QWidget (parent), ui (new Ui::IhmGroom)
00032 {
00033     qDebug() << Q_FUNC_INFO;
00034     ui->setupUi (this);
00035     setWindowTitle (NOM_APP);
00036
00037     ihmImportation = new Importation (this);
00038     messagePersonnalise = new MessagePersonnalise (this);
00039     communicationBluetooth = new CommunicationBluetooth (this);
00040     controle = new Controle (this);
00041     bdd = new Bdd (this);
00042
00043     initialiserIcôneSysteme ();
00044     connect (ui->pushButton_5, SIGNAL (clicked()), this, SLOT (
inversionImageSonnette ());
00048     // Test :
00049     connect (ui->pushButton_7, SIGNAL (clicked()), this, SLOT (testerNotification ());
00050     connect (ui->pushButton_6, SIGNAL (clicked()), this, SLOT (afficherIhmConfiguration ());
00051
00052 }
```

#### 8.4.2.2 ~IhmGroom()

```
IhmGroom::~IhmGroom ( )
```

Destructeur de la classe [IhmGroom](#).

Constructeur classe [~IhmGroom](#).

## Paramètres

<i>parent</i>	
---------------	--

Définition à la ligne 59 du fichier [ihmgroom.cpp](#).

Références [ui](#).

```
00060 {
00061     delete ui;
00062     qDebug() << Q_FUNC_INFO;
00063 }
```

## 8.4.3 Documentation des fonctions membres

## 8.4.3.1 acquitterNotification

```
IhmGroom::acquitterNotification ( ) [slot]
```

Méthode qui permet d'acquitter une notification.

Acquitter les notification.

Définition à la ligne 170 du fichier [ihmgroom.cpp](#).

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

```
00171 {
00172     qDebug() << Q_FUNC_INFO;
00173 }
```

## 8.4.3.2 afficherNotification

```
IhmGroom::afficherNotification (
    QString titre,
    QString message,
    int duree = 1000 ) [slot]
```

Méthode qui permet d'afficher une notification système.

Affichage des notification.

## Paramètres

<i>titre</i>	Le titre de la notification
<i>message</i>	Le message de la notification
<i>duree</i>	La durée en millisecondes de la notification (par défaut 5s)

Définition à la ligne 151 du fichier [ihmgroom.cpp](#).

Références [iconeSysteme](#).

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

```
00152 {
00153     QIcon icone(":/groom.png");
00154     iconeSysteme->showMessage(titre, message, icone, duree); // duree en ms
00155     /*
00156     QSystemTrayIcon::NoIcon
00157     QSystemTrayIcon::Information
00158     QSystemTrayIcon::Warning
00159     QSystemTrayIcon::Critical
00160     */
00161     //QSystemTrayIcon::MessageIcon messageIcône =
    QSystemTrayIcon::MessageIcon(QSystemTrayIcon::Information);
00162     //iconeSysteme->showMessage(titre, message, messageIcône, duree);
00163 }
```

#### 8.4.3.3 changementFonction

IhmGroom::changementFonction ( ) [slot]

change la fonction sur l'ihm

Définition à la ligne 75 du fichier [ihmgroom.cpp](#).

Références [controle](#), [Controle : :getFonction\(\)](#), et [ui](#).

Référencé par [Controle : :coderTrameAffichage\(\)](#).

```
00076 {
00077     ui->label_Fonction->setWindowTitle(controle->getFonction());
00078 }
```

#### 8.4.3.4 changementNom

IhmGroom::changementNom ( ) [slot]

change le nom sur l'ihm

Définition à la ligne 65 du fichier [ihmgroom.cpp](#).

Références [controle](#), [Controle : :getNom\(\)](#), et [ui](#).

Référencé par [Controle : :coderTrameAffichage\(\)](#).

```
00066 {
00067     ui->label_Nom->setWindowTitle(controle->getNom());
00068 }
```

#### 8.4.3.5 changementPrenom

IhmGroom::changementPrenom ( ) [slot]

change le prenom sur l'ihm

Définition à la ligne 70 du fichier [ihmgroom.cpp](#).

Références [controle](#), [Controle : :getPrenom\(\)](#), et [ui](#).

Référencé par [Controle : :coderTrameAffichage\(\)](#).

```
00071 {
00072     ui->label_Prenom->setWindowTitle(controle->getPrenom());
00073 }
```

#### 8.4.3.6 closeEvent()

```
void IhmGroom::closeEvent (
    QCloseEvent * event ) [protected]
```

Méthode redéfinie qui est appelée automatiquement lors d'une demande de fermeture.

## Paramètres

<i>event</i>	L'évènement de fermeture reçu
--------------	-------------------------------

Définition à la ligne 87 du fichier `ihmgroom.cpp`.

Références `etatInitialIcôneSysteme`, et `iconeSysteme`.

```
00088 {
00089     if (iconeSysteme->isVisible())
00090     {
00091         if (etatInitialIcôneSysteme)
00092         {
00093             QMessageBox::information(this, "Groom", "Le programme continue à s'exécuter. Utiliser le menu
00094 Quitter pour mettre fin à l'application Groom.");
00095             etatInitialIcôneSysteme = false;
00096         }
00097         hide();
00098         event->ignore();
00099     }
00100 }
```

## 8.4.3.7 initialiserIcôneSysteme()

`IhmGroom::initialiserIcôneSysteme ( ) [private]`

Méthode qui permet à l'application de s'installer dans la barre système.

Affichage dans barre des tâches.

Définition à la ligne 106 du fichier `ihmgroom.cpp`.

Références `acquitterNotification()`, `actionMaximiser`, `actionMinimiser`, `actionQuitter`, `actionRestaurer`, `etatInitialIcôneSysteme`, `iconeSysteme`, et `menuIcôneSysteme`.

Référencé par `IhmGroom()`.

```
00107 {
00108     // Crée les actions
00109     actionMinimiser = new QAction(QString::fromUtf8("Minimiser"), this);
00110     actionMaximiser = new QAction(QString::fromUtf8("Maximiser"), this);
00111     actionRestaurer = new QAction(QString::fromUtf8("Restaurer"), this);
00112     actionQuitter = new QAction(QString::fromUtf8("&Quitter"), this);
00113
00114     // Connecte les actions
00115     connect(actionMinimiser, SIGNAL(triggered(bool)), this, SLOT(hide()));
00116     connect(actionMaximiser, SIGNAL(triggered(bool)), this, SLOT(showMaximized()));
00117     connect(actionRestaurer, SIGNAL(triggered(bool)), this, SLOT(showNormal()));
00118     connect(actionQuitter, SIGNAL(triggered(bool)), qApp, SLOT(quit()));
00119
00120     // Crée le menu
00121     menuIcôneSysteme = new QMenu(this);
00122     menuIcôneSysteme->addAction(actionMinimiser);
00123     menuIcôneSysteme->addAction(actionMaximiser);
00124     menuIcôneSysteme->addAction(actionRestaurer);
00125     menuIcôneSysteme->addSeparator();
00126     menuIcôneSysteme->addAction(actionQuitter);
00127
00128     // Crée l'icône pour la barre de tâche
00129     icôneSysteme = new QSystemTrayIcon(this);
00130     icôneSysteme->setContextMenu(menuIcôneSysteme);
00131     icôneSysteme->setToolTip("Groom");
00132     QIcon icône(":/groom.png");
00133     icôneSysteme->setIcon(icône);
00134     setWindowIcon(icône);
00135
00136     connect(icôneSysteme, SIGNAL(messageClicked()), this, SLOT(
00137 acquitterNotification()));
00138     //connect (iconeSysteme, SIGNAL(activated(QSystemTrayIcon::ActivationReason)), this,
00139 SLOT(aActiveIcôneSysteme(QSystemTrayIcon::ActivationReason)));
00140
00141     icôneSysteme->show();
00142     etatInitialIcôneSysteme = true;
00143 }
```

#### 8.4.3.8 inversionImageSonnette

IhmGroom::inversionImageSonnette ( ) [slot]

Inverse l'image de la sonnette.

Définition à la ligne 201 du fichier `ihmgroom.cpp`.

Références `controle`, `Controle : :getEtatSonnette()`, `Controle : :setEtatSonnette()`, et `ui`.

Référencé par `IhmGroom()`.

```
00202 {
00203
00204     if (controle->getEtatSonnette())
00205     {
00206         ui->pushButton_5->setIcon(QIcon(":/bellDisable.png"));
00207
00208         controle->setEtatSonnette(false);
00209     }
00210
00211     else
00212     {
00213         ui->pushButton_5->setIcon(QIcon(":/bellEnable.png"));
00214         controle->setEtatSonnette(true);
00215     }
00216 }
```

#### 8.4.3.9 on\_pushButton\_2\_clicked

void IhmGroom::on\_pushButton\_2\_clicked ( ) [private], [slot]

Définition à la ligne 229 du fichier `ihmgroom.cpp`.

Références `ABSENT`, `controle`, et `Controle : :setEtatUtilisateur()`.

```
00230 {
00231     controle->setEtatUtilisateur(ABSENT);
00232 }
```

#### 8.4.3.10 on\_pushButton\_3\_clicked

IhmGroom::on\_pushButton\_3\_clicked ( ) [private], [slot]

change l'état de la personne en Libre

Définition à la ligne 219 du fichier `ihmgroom.cpp`.

Références `controle`, `LIBRE`, et `Controle : :setEtatUtilisateur()`.

```
00220 {
00221     controle->setEtatUtilisateur(LIBRE);
00222 }
```

#### 8.4.3.11 on\_pushButton\_4\_clicked

```
void IhmGroom::on_pushButton_4_clicked ( ) [private], [slot]
```

Définition à la ligne 224 du fichier [ihmgroom.cpp](#).

Références [controle](#), [OCCUPE](#), et [Controle : :setEtatUtilisateur\(\)](#).

```
00225 {  
00226     controle->setEtatUtilisateur\(OCCUPE\);  
00227 }
```

#### 8.4.3.12 on\_pushButton\_6\_clicked

```
IhmGroom::on_pushButton_6_clicked ( ) [private], [slot]
```

affiche l'ihm communication bluetooth

Définition à la ligne 181 du fichier [ihmgroom.cpp](#).

Références [communicationBluetooth](#).

```
00182 {  
00183     communicationBluetooth->exec\(\);  
00184 }
```

#### 8.4.3.13 on\_pushButton\_8\_clicked

```
IhmGroom::on_pushButton_8_clicked ( ) [private], [slot]
```

affiche l'ihm importation

change l'état de la personne en Absent

change l'état de la personne en Occupe

Définition à la ligne 186 du fichier [ihmgroom.cpp](#).

Références [ihmImportation](#).

```
00187 {  
00188     ihmImportation->exec\(\);  
00189 }
```

#### 8.4.3.14 on\_pushButton\_9\_clicked

```
void IhmGroom::on_pushButton_9_clicked ( ) [private], [slot]
```

Définition à la ligne 196 du fichier [ihmgroom.cpp](#).

Références [bdd](#).

```
00197 {  
00198     bdd->exec();  
00199 }
```

#### 8.4.3.15 on\_pushButton\_clicked

```
IhmGroom::on_pushButton_clicked ( ) [private], [slot]
```

affiche l'ihm Message personnalisé

Définition à la ligne 191 du fichier [ihmgroom.cpp](#).

Références [messagePersonnalise](#).

```
00192 {  
00193     messagePersonnalise->exec();  
00194 }
```

#### 8.4.3.16 testerNotification

```
void IhmGroom::testerNotification ( ) [slot]
```

Définition à la ligne 175 du fichier [ihmgroom.cpp](#).

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

Référencé par [IhmGroom\(\)](#).

```
00176 {  
00177     afficherNotification("Groom", "Rendez-vous annulé !");  
00178 }
```

### 8.4.4 Documentation des données membres

#### 8.4.4.1 actionMaximiser

```
QAction* IhmGroom::actionMaximiser [private]
```

L'action maximiser l'application.

Définition à la ligne 96 du fichier [ihmgroom.h](#).

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

#### 8.4.4.2 actionMinimiser

```
QAction* IhmGroom::actionMinimiser [private]
```

L'action minimiser l'application.

Définition à la ligne 95 du fichier [ihmgroom.h](#).

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

#### 8.4.4.3 actionQuitter

```
QAction* IhmGroom::actionQuitter [private]
```

L'action quitter l'application.

Définition à la ligne 98 du fichier [ihmgroom.h](#).

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

#### 8.4.4.4 actionRestaurer

```
QAction* IhmGroom::actionRestaurer [private]
```

L'action restaurer l'application.

Définition à la ligne 97 du fichier [ihmgroom.h](#).

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

#### 8.4.4.5 bdd

```
Bdd* IhmGroom::bdd [private]
```

Objet servant a controller les trame reçues et envoyés

Définition à la ligne 90 du fichier [ihmgroom.h](#).

Référencé par [IhmGroom\(\)](#), et [on\\_pushButton\\_9\\_clicked\(\)](#).

#### 8.4.4.6 communicationBluetooth

```
CommunicationBluetooth* IhmGroom::communicationBluetooth [private]
```

L'interface de la Configuration.

Définition à la ligne 86 du fichier [ihmgroom.h](#).

Référencé par [IhmGroom\(\)](#), et [on\\_pushButton\\_6\\_clicked\(\)](#).

#### 8.4.4.7 controle

```
Controle* IhmGroom::controle [private]
```

Définition à la ligne 89 du fichier [ihmgroom.h](#).

Référencé par [changementFonction\(\)](#), [changementNom\(\)](#), [changementPrenom\(\)](#), [IhmGroom\(\)](#), [inversionImageSonnette\(\)](#), [on\\_pushButton\\_2\\_clicked\(\)](#), [on\\_pushButton\\_3\\_clicked\(\)](#), et [on\\_pushButton\\_4\\_clicked\(\)](#).

#### 8.4.4.8 etatInitialIcôneSysteme

```
bool IhmGroom::etatInitialIcôneSysteme [private]
```

Booléen indiquant si c'est la première demande Quitter.

Définition à la ligne 99 du fichier [ihmgroom.h](#).

Référencé par [closeEvent\(\)](#), et [initialiserIcôneSysteme\(\)](#).

#### 8.4.4.9 icôneSysteme

```
QSystemTrayIcon* IhmGroom::icôneSysteme [private]
```

L'icône de l'application pour la barre système.

Définition à la ligne 93 du fichier [ihmgroom.h](#).

Référencé par [afficherNotification\(\)](#), [closeEvent\(\)](#), et [initialiserIcôneSysteme\(\)](#).

#### 8.4.4.10 ihmImportation

```
Importation* IhmGroom::ihmImportation [private]
```

L'interface de l'[Importation](#).

Définition à la ligne 87 du fichier [ihmgroom.h](#).

Référencé par [IhmGroom\(\)](#), et [on\\_pushButton\\_8\\_clicked\(\)](#).

#### 8.4.4.11 menuIcôneSysteme

```
QMenu* IhmGroom::menuIcôneSysteme [private]
```

Le menu de l'application.

Définition à la ligne 94 du fichier [ihmgroom.h](#).

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

## 8.4.4.12 messagePersonnalise

```
MessagePersonnalise* IhmGroom::messagePersonnalise [private]
```

L'interface du Message Personnalisé

Définition à la ligne 88 du fichier [ihmgroom.h](#).

Référencé par [IhmGroom\(\)](#), et [on\\_pushButton\\_clicked\(\)](#).

## 8.4.4.13 ui

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

L'interface utilisateur.

Définition à la ligne 85 du fichier [ihmgroom.h](#).

Référencé par [changementFonction\(\)](#), [changementNom\(\)](#), [changementPrenom\(\)](#), [IhmGroom\(\)](#), [inversionImageSonnette\(\)](#), et [~IhmGroom\(\)](#).

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

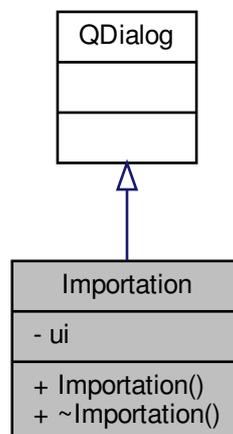
- [ihmgroom.h](#)
- [ihmgroom.cpp](#)

## 8.5 Référence de la classe Importation

Classe permettant de d'importer le calendrier.

```
#include "importation.h"
```

Graphe de collaboration de Importation :



## Fonctions membres publiques

- [Importation](#) ([QWidget](#) \*parent=nullptr)  
*Constructeur de la classe [Importation](#).*
- [~Importation](#) ()  
*Destructeur de la classe [Importation](#).*

## Attributs privés

- `Ui : :Importation * ui`

### 8.5.1 Description détaillée

Classe permettant de d'importer le calendrier.

Définition à la ligne 17 du fichier [importation.h](#).

### 8.5.2 Documentation des constructeurs et destructeur

#### 8.5.2.1 Importation()

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

Constructeur de la classe [Importation](#).

#### Paramètres

<i>parent</i>	
---------------	--

Définition à la ligne 4 du fichier [importation.cpp](#).

Références [ui](#).

```
00004                                     : QDialog(parent), ui(new Ui::Importation)
00005 {
00006     QDebug() << Q\_FUNC\_INFO;
00007     ui->setupUi(this);
00008 }
```

#### 8.5.2.2 ~Importation()

```
Importation::~Importation ( )
```

Destructeur de la classe [Importation](#).

#### Paramètres

<i>parent</i>	
---------------	--

Définition à la ligne 10 du fichier [importation.cpp](#).

Références [ui](#).

```
00011 {
00012     delete ui;
00013     qDebug() << Q_FUNC_INFO;
00014 }
```

### 8.5.3 Documentation des données membres

#### 8.5.3.1 ui

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

Définition à la ligne 38 du fichier [importation.h](#).

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

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

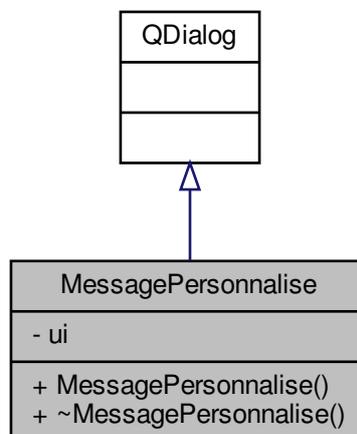
- [importation.h](#)
- [importation.cpp](#)

## 8.6 Référence de la classe MessagePersonnalise

Classe permettant de personnaliser son message.

```
#include "messagepersonnalise.h"
```

Graphe de collaboration de MessagePersonnalise :



## Fonctions membres publiques

- [MessagePersonnalise](#) ([QWidget](#) \*parent=nullptr)  
Constructeur de la classe [MessagePersonnalise](#).
- [~MessagePersonnalise](#) ()  
Destructeur de la classe [MessagePersonnalise](#).

## Attributs privés

- `Ui : :MessagePersonnalise * ui`

### 8.6.1 Description détaillée

Classe permettant de personnaliser son message.

Définition à la ligne 16 du fichier [messagepersonnalise.h](#).

### 8.6.2 Documentation des constructeurs et destructeur

#### 8.6.2.1 MessagePersonnalise()

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

Constructeur de la classe [MessagePersonnalise](#).

#### Paramètres

<i>parent</i>
---------------

Définition à la ligne 4 du fichier [messagepersonnalise.cpp](#).

Références [ui](#).

```
00004
00005     QDialog(parent),
00006     ui(new Ui::MessagePersonnalise)
00007 {
00008     qDebug() << Q_FUNC_INFO;
00009     ui->setupUi(this);
00010 }
```

#### 8.6.2.2 ~MessagePersonnalise()

```
MessagePersonnalise::~MessagePersonnalise ( )
```

Destructeur de la classe [MessagePersonnalise](#).

## Paramètres

<i>parent</i>
---------------

Définition à la ligne 12 du fichier [messagepersonnalise.cpp](#).

Références [ui](#).

```
00013 {  
00014     delete ui;  
00015     qDebug() << Q_FUNC_INFO;  
00016 }
```

### 8.6.3 Documentation des données membres

#### 8.6.3.1 ui

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

Définition à la ligne 37 du fichier [messagepersonnalise.h](#).

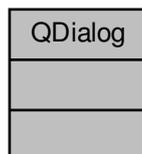
Référencé par [MessagePersonnalise\(\)](#), et [~MessagePersonnalise\(\)](#).

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

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

## 8.7 Référence de la classe QDialog

Grphe de collaboration de QDialog :

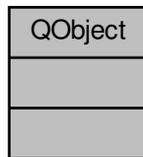


La documentation de cette classe a été générée à partir du fichier suivant :

- [messagepersonnalise.h](#)

## 8.8 Référence de la classe QObject

Graphe de collaboration de QObject :

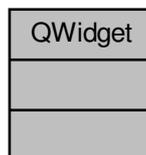


La documentation de cette classe a été générée à partir du fichier suivant :

— [controle.h](#)

## 8.9 Référence de la classe QWidget

Graphe de collaboration de QWidget :



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

— [ihmgroom.h](#)

## 9 Documentation des fichiers

### 9.1 Référence du fichier bdd.cpp

```
#include "bdd.h"
#include "ui_bdd.h"
```

## 9.2 bdd.cpp

```

00001 #include "bdd.h"
00002 #include "ui_bdd.h"
00003
00004 Bdd::Bdd(QWidget *parent) :
00005     QDialog(parent),
00006     ui(new Ui::Bdd)
00007 {
00008     ui->setupUi(this);
00009 }
00010
00011 Bdd::~Bdd()
00012 {
00013     delete ui;
00014 }

```

## 9.3 Référence du fichier bdd.h

```
#include <QtWidgets>
```

### Classes

— class [Bdd](#)

### Espaces de nommage

— [Ui](#)

## 9.4 bdd.h

```

00001 #ifndef BDD_H
00002 #define BDD_H
00003
00004 #include <QtWidgets>
00005
00006 namespace Ui {
00007 class Bdd;
00008 }
00009
00010 class Bdd : public QDialog
00011 {
00012     Q_OBJECT
00013
00014 public:
00015     explicit Bdd(QWidget *parent = nullptr);
00016     ~Bdd();
00017
00018 private:
00019     Ui::Bdd *ui;
00020 };
00021
00022 #endif // BDD_H

```

## 9.5 Référence du fichier Changelog.md

## 9.6 Changelog.md

```

00001 \page page_changelog Changelog
00002
00003 r1 | www-data | 2020-02-01 15:03:29 +0100 (sam. 01 févr. 2020) | 1 ligne
00004
00005 Creating initial repository structure

```

## 9.7 Référence du fichier communicationbluetooth.cpp

```
#include "communicationbluetooth.h"
#include "ui_communicationbluetooth.h"
```

## 9.8 communicationbluetooth.cpp

```
00001 #include "communicationbluetooth.h"
00002 #include "ui_communicationbluetooth.h"
00003
00004 CommunicationBluetooth::CommunicationBluetooth(
00005     QWidget *parent) :
00006     QDialog(parent),
00007     ui(new Ui::CommunicationBluetooth)
00008 {
00009     qDebug() << Q_FUNC_INFO;
00010     ui->setupUi(this);
00011 }
00012 CommunicationBluetooth::~CommunicationBluetooth()
00013 {
00014     delete ui;
00015     qDebug() << Q_FUNC_INFO;
00016 }
```

## 9.9 Référence du fichier communicationbluetooth.h

```
#include <QtWidgets>
```

### Classes

- class [CommunicationBluetooth](#)  
*Class permettant de mettre en place une communication bluetooth.*

### Espaces de nommage

- [Ui](#)

## 9.10 communicationbluetooth.h

```
00001 #ifndef COMMUNICATIONBLUETOOTH_H
00002 #define COMMUNICATIONBLUETOOTH_H
00003
00004 #include <QtWidgets>
00005
00006 class IhmGroom;
00007 namespace Ui {
00008 class CommunicationBluetooth;
00009 }
00010
00011 class CommunicationBluetooth : public QDialog
00012 {
00013     Q_OBJECT
00014
00015 public:
00016     explicit CommunicationBluetooth(QWidget *parent = nullptr);
00017     ~CommunicationBluetooth();
00018
00019 private:
00020     Ui::CommunicationBluetooth *ui;
00021 };
00022 #endif // COMMUNICATIONBLUETOOTH_H
```

## 9.11 Référence du fichier controle.cpp

```
#include "controle.h"  
#include <QDebug>
```

## 9.12 controle.cpp

```
00001 #include "controle.h"  
00002 #include <QDebug>  
00003  
00004  
00005 Controle::Controle(QObject *parent) : QObject(parent)  
00006 {  
00007     qDebug() << Q_FUNC_INFO;  
00008     //     ui = new IhmGroom;  
00009     etatSonnette = true;  
00010  
00011 }  
00012  
00013 Controle::~Controle()  
00014 {  
00015     qDebug() << Q_FUNC_INFO;  
00016 }  
00017  
00018 void Controle::setEtatUtilisateur(int newEtatUtilisateur)  
00019 {  
00020     etatUtilisateur = newEtatUtilisateur;  
00021 }  
00022  
00023 int Controle::getEtatUtilisateur()  
00024 {  
00025     return etatUtilisateur;  
00026 }  
00027  
00028 void Controle::setEtatSonnette(bool newEtatSonnette)  
00029 {  
00030     etatSonnette = newEtatSonnette;  
00031 }  
00032  
00033 bool Controle::getEtatSonnette()  
00034 {  
00035     return etatSonnette;  
00036 }  
00037  
00038 void Controle::setPresence(bool newPresence)  
00039 {  
00040     presence = newPresence;  
00041 }  
00042  
00043 bool Controle::getPresence()  
00044 {  
00045     return presence;  
00046 }  
00047  
00048  
00049 void Controle::decoderTrameGroom(QString trameGroom)  
00050 {  
00051  
00052     setEtatUtilisateur((trameGroom.section(";",1,1)).toInt());  
00053  
00054  
00055     if (trameGroom.section(";",2,2) == 1)  
00056     {  
00057         setEtatSonnette(true);  
00058     }  
00059     else  
00060     {  
00061         setEtatSonnette(false);  
00062     }  
00063  
00064     if (trameGroom.section(";",3,3) == 1)  
00065     {  
00066         setPresence(true);  
00067     }  
00068     else  
00069     {  
00070         setPresence(false);  
00071     }  
00072  
00073 }  
00074  
00075 void Controle::setNom(QString newNom)
```

```

00076 {
00077     nom = newNom;
00078 }
00079
00080 QString Controle::getNom()
00081 {
00082     return nom;
00083 }
00084 void Controle::setPrenom(QString newPrenom)
00085 {
00086     prenom = newPrenom;
00087 }
00088 QString Controle::getPrenom()
00089 {
00090     return prenom;
00091 }
00092 void Controle::setFonction(QString newFonction)
00093 {
00094     fonction = newFonction;
00095 }
00096 QString Controle::getFonction()
00097 {
00098     return fonction;
00099 }
00100
00101 void Controle::coderTrameAffichage(QString
trameAffichage)
00102 {
00103     setNom(trameAffichage.section(";",1,1));
00104
00105     setPrenom(trameAffichage.section(";",2,2));
00106
00107     setFonction(trameAffichage.section(";",3,3));
00108
00109     ui->changementNom();
00110     ui->changementPrenom();
00111     ui->changementFonction();
00112 }

```

### 9.13 Référence du fichier controle.h

```

#include <QObject>
#include <QDateTime>
#include <QString>
#include "communicationbluetooth.h"
#include "ihmgroom.h"
#include "messagepersonnalise.h"

```

#### Classes

- class **Controle**  
*Classe assurant le controle et la validé de l'envoi et réception des trames.*

### 9.14 controle.h

```

00001 #ifndef CONTROLE_H
00002 #define CONTROLE_H
00003
00004 #include <QObject>
00005 #include <QDateTime>
00006 #include <QString>
00007 #include "communicationbluetooth.h"
00008 #include "ihmgroom.h"
00009 #include "messagepersonnalise.h"
00010
00017 class IhmGroom;
00018 class CommunicationBluetooth;
00019 class MessagePersonnalise;
00020
00021
00022 class Controle : public QObject
00023 {
00024     Q_OBJECT

```

```

00025
00026     private:
00027         CommunicationBluetooth *communicationBluetooth;
00028         MessagePersonnalise *messagePersonnalise;
00029         IhmGroom *ui;
00031         QString trameGroom;
00032         QString trameAffichage;
00033         QString trameMsgPerso;
00042         bool verifierChecksum(QString trameGroom);
00048         void decoderTrameGroom(QString trameGroom);
00054         void coderTrameGroom(QString trameGroom);
00060         void coderTrameAffichage(QString trameAffichage);
00066         void coderTrameMsgPerso(QString trameMsgPerso);
00067
00068
00069         int etatUtilisateur;
00070         bool etatSonnette;
00071         bool presence;
00073         QString nom;
00074         QString prenom;
00075         QString fonction;
00079     public:
00086         Controle(QObject *parent = nullptr);
00093         ~Controle();
00100         void setEtatSonnette(bool newEtatSonnette);
00107         bool getEtatSonnette();
00114         void setEtatUtilisateur(int newEtatUtilisateur);
00121         int getEtatUtilisateur();
00128         void setPresence(bool newPresence);
00135         bool getPresence();
00136
00143         void setNom(QString newNom);
00150         QString getNom();
00157         void setPrenom(QString newPrenom);
00164         QString getPrenom();
00171         void setFonction(QString newFonction);
00178         QString getFonction();
00179
00180
00181     public slots:
00182
00183 };
00184
00185 #endif // CONTROLE_H

```

## 9.15 Référence du fichier ihmgroom.cpp

Définition de la classe [IhmGroom](#).

```

#include "ihmgroom.h"
#include "ui_ihmgroom.h"
#include "communicationbluetooth.h"
#include "ui_communicationbluetooth.h"
#include "importation.h"
#include "ui_importation.h"
#include "messagepersonnalise.h"
#include "ui_messagepersonnalise.h"
#include "controle.h"
#include "bdd.h"
#include "ui_bdd.h"

```

### 9.15.1 Description détaillée

Définition de la classe [IhmGroom](#).

#### Auteur

Etienne Valette

Définition dans le fichier [ihmgroom.cpp](#).

## 9.16 ihmgroom.cpp

```

00001 #include "ihmgroom.h"
00002 #include "ui_ihmgroom.h"
00003 #include "communicationbluetooth.h"
00004 #include "ui_communicationbluetooth.h"
00005 #include "importation.h"
00006 #include "ui_importation.h"
00007 #include "messagepersonnalise.h"
00008 #include "ui_messagepersonnalise.h"
00009 #include "controle.h"
00010 #include "bdd.h"
00011 #include "ui_bdd.h"
00012
00031 IhmGroom::IhmGroom(QWidget *parent) : QWidget(parent), ui(new
    Ui::IhmGroom)
00032 {
00033     qDebug() << Q_FUNC_INFO;
00034     ui->setupUi(this);
00035     setWindowTitle(NOM_APP);
00036
00037     ihmImportation = new Importation(this);
00038     messagePersonnalise = new MessagePersonnalise(this);
00039     communicationBluetooth = new CommunicationBluetooth(this);
00040     controle = new Controle(this);
00041     bdd = new Bdd(this);
00042
00043     initialiserIcôneSysteme();
00044     connect(ui->pushButton_5, SIGNAL(clicked()), this, SLOT(
        inversionImageSonnette()));
00048     // Test :
00049     connect(ui->pushButton_7, SIGNAL(clicked()), this, SLOT(testerNotification()));
00050     connect(ui->pushButton_6, SIGNAL(clicked()), this, SLOT(afficherIhmConfiguration()));
00051
00052 }
00053
00059 IhmGroom::~IhmGroom()
00060 {
00061     delete ui;
00062     qDebug() << Q_FUNC_INFO;
00063 }
00064
00065 void IhmGroom::changementNom()
00066 {
00067     ui->label_Nom->setWindowTitle(controle->getNom());
00068 }
00069
00070 void IhmGroom::changementPrenom()
00071 {
00072     ui->label_Prenom->setWindowTitle(controle->getPrenom());
00073 }
00074
00075 void IhmGroom::changementFonction()
00076 {
00077     ui->label_Fonction->setWindowTitle(controle->getFonction());
00078 }
00079
00080
00087 void IhmGroom::closeEvent(QCloseEvent *event)
00088 {
00089     if (iconeSysteme->isVisible())
00090     {
00091         if(etatInitialIcôneSysteme)
00092         {
00093             QMessageBox::information(this, "Groom", "Le programme continue à s'exécuter. Utiliser le menu
                Quitter pour mettre fin à l'application Groom.");
00094             etatInitialIcôneSysteme = false;
00095         }
00096         hide();
00097         event->ignore();
00098     }
00099 }
00100
00106 void IhmGroom::initialiserIcôneSysteme()
00107 {
00108     // Crée les actions
00109     actionMinimiser = new QAction(QString::fromUtf8("Minimiser"), this);
00110     actionMaximiser = new QAction(QString::fromUtf8("Maximiser"), this);
00111     actionRestaurer = new QAction(QString::fromUtf8("Restaurer"), this);
00112     actionQuitter = new QAction(QString::fromUtf8("&Quitter"), this);
00113
00114     // Connecte les actions
00115     connect(actionMinimiser, SIGNAL(triggered(bool)), this, SLOT(hide()));
00116     connect(actionMaximiser, SIGNAL(triggered(bool)), this, SLOT(showMaximized()));
00117     connect(actionRestaurer, SIGNAL(triggered(bool)), this, SLOT(showNormal()));
00118     connect(actionQuitter, SIGNAL(triggered(bool)), qApp, SLOT(quit()));
00119
00120     // Crée le menu
00121     menuIcôneSysteme = new QMenu(this);
00122     menuIcôneSysteme->addAction(actionMinimiser);

```

```

00123     menuIcôneSysteme->addAction(actionMaximiser);
00124     menuIcôneSysteme->addAction(actionRestaurer);
00125     menuIcôneSysteme->addSeparator();
00126     menuIcôneSysteme->addAction(actionQuitter);
00127
00128     // Crée l'icône pour la barre de tâche
00129     icôneSysteme = new QSystemTrayIcon(this);
00130     icôneSysteme->setContextMenu(menuIcôneSysteme);
00131     icôneSysteme->setToolTip("Groom");
00132     QIcon icône(":/groom.png");
00133     icôneSysteme->setIcon(icône);
00134     setWindowIcon(icône);
00135
00136     connect(icôneSysteme, SIGNAL(messageClicked()), this, SLOT(
acquitterNotification()));
00137     //connect(icôneSysteme, SIGNAL(activated(QSystemTrayIcon::ActivationReason)), this,
SLOT(aActiveIcôneSysteme(QSystemTrayIcon::ActivationReason)));
00138
00139     icôneSysteme->show();
00140     etatInitialIcôneSysteme = true;
00141 }
00142
00151 void IhmGroom::afficherNotification(QString titre, QString message, int duree
)
00152 {
00153     QIcon icône(":/groom.png");
00154     icôneSysteme->showMessage(titre, message, icône, duree); // duree en ms
00155     /*
00156     QSystemTrayIcon::NoIcon
00157     QSystemTrayIcon::Information
00158     QSystemTrayIcon::Warning
00159     QSystemTrayIcon::Critical
00160     */
00161     //QSystemTrayIcon::MessageIcon messageIcône =
QSystemTrayIcon::MessageIcon(QSystemTrayIcon::Information);
00162     //icôneSysteme->showMessage(titre, message, messageIcône, duree);
00163 }
00164
00170 void IhmGroom::acquitterNotification()
00171 {
00172     qDebug() << Q_FUNC_INFO;
00173 }
00174
00175 void IhmGroom::testerNotification()
00176 {
00177     afficherNotification("Groom", "Rendez-vous annulé !");
00178 }
00179
00180
00181 void IhmGroom::on_pushButton_6_clicked()
00182 {
00183     communicationBluetooth->exec();
00184 }
00185
00186 void IhmGroom::on_pushButton_8_clicked()
00187 {
00188     ihmImportation->exec();
00189 }
00190
00191 void IhmGroom::on_pushButton_clicked()
00192 {
00193     messagePersonnalise->exec();
00194 }
00195
00196 void IhmGroom::on_pushButton_9_clicked()
00197 {
00198     bdd->exec();
00199 }
00200
00201 void IhmGroom::inversionImageSonnette()
00202 {
00203
00204     if (controle->getEtatSonnette())
00205     {
00206         ui->pushButton_5->setIcon(QIcon(":/bellDisable.png"));
00207
00208         controle->setEtatSonnette(false);
00209     }
00210     else
00211     {
00212         ui->pushButton_5->setIcon(QIcon(":/bellEnable.png"));
00213         controle->setEtatSonnette(true);
00214     }
00215 }
00216 }
00217
00218
00219 void IhmGroom::on_pushButton_3_clicked()
00220 {
00221     controle->setEtatUtilisateur(LIBRE);
00222 }

```

```
00223
00224 void IhmGroom::on_pushButton_4_clicked()
00225 {
00226     controle->setEtatUtilisateur(OCCUPE);
00227 }
00228
00229 void IhmGroom::on_pushButton_2_clicked()
00230 {
00231     controle->setEtatUtilisateur(ABSENT);
00232 }
00233
00234
00235
00236
```

## 9.17 Référence du fichier ihmgroom.h

Déclaration de la classe [IhmGroom](#).

```
#include <QtWidgets>
#include <QIcon>
```

### Classes

- class [IhmGroom](#)  
*Déclaration de la classe [IhmGroom](#).*

### Espaces de nommage

- [Ui](#)

### Macros

- #define [ABSENT](#) 1  
*Constante contenant le numéro 1.*
- #define [LIBRE](#) 0  
*Constante contenant le numéro 0.*
- #define [NOM\\_APP](#) "Groom"  
*Constante contenant le début de la trame sonde selon le protocole.*
- #define [OCCUPE](#) 2  
*Constante contenant le numéro 2.*

#### 9.17.1 Description détaillée

Déclaration de la classe [IhmGroom](#).

#### Auteur

Etienne Valette

Définition dans le fichier [ihmgroom.h](#).

#### 9.17.2 Documentation des macros

### 9.17.2.1 ABSENT

```
#define ABSENT 1
```

Constante contenant le numéro 1.

Définition à la ligne 30 du fichier [ihmgroom.h](#).

Référencé par [IhmGroom](#) : [:on\\_pushButton\\_2\\_clicked\(\)](#).

### 9.17.2.2 LIBRE

```
#define LIBRE 0
```

Constante contenant le numéro 0.

Définition à la ligne 24 du fichier [ihmgroom.h](#).

Référencé par [IhmGroom](#) : [:on\\_pushButton\\_3\\_clicked\(\)](#).

### 9.17.2.3 NOM\_APP

```
#define NOM_APP "Groom"
```

Constante contenant le début de la trame sonde selon le protocole.

Définition à la ligne 18 du fichier [ihmgroom.h](#).

Référencé par [IhmGroom](#) : [:IhmGroom\(\)](#).

### 9.17.2.4 OCCUPE

```
#define OCCUPE 2
```

Constante contenant le numéro 2.

Définition à la ligne 36 du fichier [ihmgroom.h](#).

Référencé par [IhmGroom](#) : [:on\\_pushButton\\_4\\_clicked\(\)](#).

## 9.18 ihmgroom.h

```

00001 #ifndef IHMGROOM_H
00002 #define IHMGROOM_H
00003
00011 #include <QtWidgets>
00012 #include <QIcon>
00018 #define NOM_APP "Groom"
00019
00024 #define LIBRE 0
00025
00030 #define ABSENT 1
00031
00036 #define OCCUPE 2
00037
00038
00039
00040 class CommunicationBluetooth;
00041 class Importation;
00042 class MessagePersonnalise;
00043 class Controle;
00044 class Bdd;
00045
00046 namespace Ui {
00047 class IhmGroom;
00048 }
00049
00054 class IhmGroom : public QWidget
00055 {
00056     Q_OBJECT
00057
00058 public:
00065     explicit IhmGroom(QWidget *parent = nullptr);
00072     ~IhmGroom();
00073
00074 protected:
00081     void closeEvent(QCloseEvent *event);
00082
00083 private:
00084
00085     Ui::IhmGroom *ui;
00086     CommunicationBluetooth *communicationBluetooth;
00087     Importation *ihmImportation;
00088     MessagePersonnalise *messagePersonnalise;
00089     Controle *controle;
00090     Bdd *bdd;
00091
00092
00093     QSystemTrayIcon *iconeSysteme;
00094     QMenu *menuIconeSysteme;
00095     QAction *actionMinimiser;
00096     QAction *actionMaximiser;
00097     QAction *actionRestaurer;
00098     QAction *actionQuitter;
00099     bool etatInitialIconeSysteme;
00100
00105     void initialiserIconeSysteme();
00106
00107 public slots:
00113     void afficherNotification(QString titre, QString message, int duree=1000);
00119     void acquitterNotification();
00120
00121     // Test
00122     void testerNotification();
00128     void inversionImageSonnette();
00134     void changementNom();
00140     void changementPrenom();
00146     void changementFonction();
00147
00148 private slots:
00154     void on_pushButton_6_clicked();
00160     void on_pushButton_8_clicked();
00166     void on_pushButton_clicked();
00172     void on_pushButton_3_clicked();
00178     void on_pushButton_4_clicked();
00184     void on_pushButton_2_clicked();
00185     void on_pushButton_9_clicked();
00186 };
00187
00188 #endif // IHMGROOM_H

```

## 9.19 Référence du fichier importation.cpp

```

#include "importation.h"
#include "ui_importation.h"

```

## 9.20 importation.cpp

```

00001 #include "importation.h"
00002 #include "ui_importation.h"
00003
00004 Importation::Importation(QWidget *parent) :
    QDialog(parent), ui(new Ui::Importation)
00005 {
00006     qDebug() << Q_FUNC_INFO;
00007     ui->setupUi(this);
00008 }
00009
00010 Importation::~Importation()
00011 {
00012     delete ui;
00013     qDebug() << Q_FUNC_INFO;
00014 }

```

## 9.21 Référence du fichier importation.h

```
#include <QtWidgets>
```

### Classes

- class [Importation](#)  
*Classe permettant de d'importer le calendrier.*

### Espaces de nommage

- [Ui](#)

## 9.22 importation.h

```

00001 #ifndef IMPORTATION_H
00002 #define IMPORTATION_H
00003
00004 #include <QtWidgets>
00005
00012 class IhmGroom;
00013 namespace Ui {
00014 class Importation;
00015 }
00016
00017 class Importation : public QDialog
00018 {
00019     Q_OBJECT
00020
00021 public:
00028     explicit Importation(QWidget *parent = nullptr);
00035     ~Importation();
00036
00037 private:
00038     Ui::Importation *ui;
00039 };
00040
00041 #endif // IMPORTATION_H

```

## 9.23 Référence du fichier main.cpp

Programme principal GroomQT.

```

#include "ihmgroom.h"
#include <QApplication>

```

## Fonctions

— int `main` (int argc, char \*argv[])

### 9.23.1 Description détaillée

Programme principal GroomQT.

Crée et affiche la fenêtre principale de l'application GroomQT

#### Auteur

Etienne Valette

Définition dans le fichier `main.cpp`.

### 9.23.2 Documentation des fonctions

#### 9.23.2.1 main()

```
main (
    int argc,
    char * argv[] )
```

#### Paramètres

<code>argc</code>	
<code>argv[]</code>	

#### Renvoie

int

Définition à la ligne 19 du fichier `main.cpp`.

```
00020 {
00021     QApplication a(argc, argv);
00022     IhmGroom w;
00023     w.show();
00024
00025     return a.exec();
00026 }
```

## 9.24 main.cpp

```
00001 #include "ihmgroom.h"
00002 #include <QApplication>
00003
00019 int main(int argc, char *argv[])
00020 {
00021     QApplication a(argc, argv);
00022     IhmGroom w;
00023     w.show();
00024
00025     return a.exec();
00026 }
```

## 9.25 Référence du fichier messagepersonnalise.cpp

```
#include "messagepersonnalise.h"
#include "ui_messagepersonnalise.h"
```

## 9.26 messagepersonnalise.cpp

```
00001 #include "messagepersonnalise.h"
00002 #include "ui_messagepersonnalise.h"
00003
00004 MessagePersonnalise::MessagePersonnalise(
    QWidget *parent) :
00005     QDialog(parent),
00006     ui(new Ui::MessagePersonnalise)
00007 {
00008     qDebug() << Q_FUNC_INFO;
00009     ui->setupUi(this);
00010 }
00011
00012 MessagePersonnalise::~MessagePersonnalise()
00013 {
00014     delete ui;
00015     qDebug() << Q_FUNC_INFO;
00016 }
```

## 9.27 Référence du fichier messagepersonnalise.h

```
#include <QtWidgets>
```

## Classes

- class [MessagePersonnalise](#)  
*Classe permettant de personnaliser son message.*

## Espaces de nommage

- [Ui](#)

## 9.28 messagepersonnalise.h

```
00001 #ifndef MESSAGEPERSONNALISE_H
00002 #define MESSAGEPERSONNALISE_H
00003
00004 #include <QtWidgets>
00005
00012 namespace Ui {
00013 class MessagePersonnalise;
00014 }
00015
00016 class MessagePersonnalise : public QDialog
00017 {
00018     Q_OBJECT
00019
00020 public:
00027     explicit MessagePersonnalise(QWidget *parent = nullptr);
00034     ~MessagePersonnalise();
00035
00036 private:
00037     Ui::MessagePersonnalise *ui;
00038 };
00039
00040 #endif // MESSAGEPERSONNALISE_H
```

## 9.29 Référence du fichier README.md

## 9.30 README.md

```

00001 \mainpage Le projet
00002
00003 \tableofcontents
00004
00005 Le portier connecté « grOOm » permettra à l'occupant du bureau de communiquer sa disponibilité avec
des personnes extérieures (visiteurs). Pour cela, il utilise une application soit en version PC soit mobile.
00006
00007 \section section_tdm Table des matières
00008 - \ref page_README
00009 - \ref page_changelog
00010 \if todo
00011 - \ref todo
00012 \endif
00013 \if bug
00014 - \ref bug
00015 \endif
00016 - \ref page_about
00017 - \ref page_licence
00018
00019 \section section_infos Informations
00020
00021 \author Etienne Valette <valette.etienne@gmail.com>
00022 \date 2020
00023 \version 0.2
00024 \see https://svn.riouxsvn.com/groom/
00025
00026
00027 \page page_README README
00028
00029 [TOC]
00030
00031 # Projet {#projet}
00032
00033 ## Présentation {#presentation}
00034
00035 Le portier connecté « grOOm » permettra à l'occupant du bureau de communiquer sa disponibilité avec
des personnes extérieures (visiteurs). Pour cela, il utilise une application soit en version PC soit mobile.
00036
00037 Tout en s'intégrant facilement à l'environnement, il résout le manque d'interface entre les
utilisateurs et les bureaux permettant de travailler plus efficacement.
00038
00039 A partir de l'application, l'occupant informe le visiteur de son état : "Libre", "Occupé" ou "Absent".
Il aura la possibilité d'ajouter un message "libre" qui s'affichera alors sur l'écran du portier. S'il le
souhaite, il informera le visiteur que celui-ci peut "Entrer" (ou "Entrez"). L'occupant dispose d'un mode
"Sonnette" qu'il peut activer sur le portier connecté. Dans ce mode, le visiteur pourra sonner via l'écran
tactile et prévenir l'occupant de sa présence.
00040
00041 Version : PC Desktop Qt
00042
00043 ## Informations {#informations}
00044
00045 \author Etienne Valette <valette.etienne@gmail.com>
00046 \date 2020
00047 \version 0.2
00048 \see https://svn.riouxsvn.com/groom/
00049
00050
00051 \page page_about A propos
00052
00053 \author Etienne Valette <valette.etienne@gmail.com>
00054 \date 2020
00055 \version 0.2
00056 \see https://svn.riouxsvn.com/groom/
00057
00058
00059 \page page_licence Licence GPL
00060
00061 This program is free software; you can redistribute it and/or modify
00062 it under the terms of the GNU General Public License as published by
00063 the Free Software Foundation; either version 2 of the License, or
00064 (at your option) any later version.
00065
00066 This program is distributed in the hope that it will be useful,
00067 but WITHOUT ANY WARRANTY; without even the implied warranty of
00068 MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
00069 GNU General Public License for more details.
00070
00071 You should have received a copy of the GNU General Public License
00072 along with this program; if not, write to the Free Software
00073 Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

```