

Borne de recharge C++

V2.0

Generated by Doxygen 1.8.17

| | |
|--|----------|
| 1 Class Index | 1 |
| 1.1 Class List | 1 |
| 2 File Index | 3 |
| 2.1 File List | 3 |
| 3 Class Documentation | 5 |
| 3.1 BaseDonnee Class Reference | 5 |
| 3.1.1 Constructor & Destructor Documentation | 6 |
| 3.1.1.1 BaseDonnee() [1/2] | 6 |
| 3.1.1.2 BaseDonnee() [2/2] | 6 |
| 3.1.2 Member Function Documentation | 6 |
| 3.1.2.1 afficher_liste_client() [1/2] | 6 |
| 3.1.2.2 afficher_liste_client() [2/2] | 6 |
| 3.1.2.3 ajouter_client() [1/2] | 6 |
| 3.1.2.4 ajouter_client() [2/2] | 7 |
| 3.1.2.5 authentifier() [1/2] | 7 |
| 3.1.2.6 authentifier() [2/2] | 7 |
| 3.1.2.7 ecriture_clients_fichier() | 8 |
| 3.1.2.8 init_base_client() [1/2] | 8 |
| 3.1.2.9 init_base_client() [2/2] | 8 |
| 3.1.2.10 re_authentifier() [1/2] | 8 |
| 3.1.2.11 re_authentifier() [2/2] | 9 |
| 3.1.2.12 supprimer_client() [1/2] | 9 |
| 3.1.2.13 supprimer_client() [2/2] | 9 |
| 3.1.3 Member Data Documentation | 9 |
| 3.1.3.1 base_donnes [1/2] | 9 |
| 3.1.3.2 base_donnes [2/2] | 10 |
| 3.1.3.3 dernier_client | 10 |
| 3.1.3.4 i | 10 |
| 3.1.3.5 nb_clients | 10 |
| 3.2 Borne Class Reference | 11 |
| 3.2.1 Constructor & Destructor Documentation | 11 |
| 3.2.1.1 Borne() | 11 |
| 3.3 Boutons Class Reference | 11 |
| 3.3.1 Constructor & Destructor Documentation | 12 |
| 3.3.1.1 Boutons() | 12 |
| 3.3.2 Member Function Documentation | 12 |
| 3.3.2.1 charge() | 12 |
| 3.3.2.2 stop() | 12 |
| 3.3.3 Member Data Documentation | 13 |
| 3.3.3.1 depart_timer | 13 |
| 3.3.3.2 io | 13 |

| | | |
|---------|--|----|
| 3.4 | Generateur Class Reference | 13 |
| 3.4.1 | Constructor & Destructor Documentation | 14 |
| 3.4.1.1 | Generateur() | 14 |
| 3.4.2 | Member Function Documentation | 14 |
| 3.4.2.1 | charger() | 14 |
| 3.4.2.2 | charger_batterie() | 15 |
| 3.4.2.3 | deconnecter() | 15 |
| 3.4.2.4 | fermer_AC() | 15 |
| 3.4.2.5 | generateursave_initialiser() | 16 |
| 3.4.2.6 | generer_PWM() | 16 |
| 3.4.2.7 | ouvrir_AC() | 16 |
| 3.4.2.8 | tension() | 17 |
| 3.4.3 | Member Data Documentation | 17 |
| 3.4.3.1 | depart_timer | 17 |
| 3.4.3.2 | io | 17 |
| 3.4.3.3 | num_client | 18 |
| 3.5 | LecteurCarte Class Reference | 18 |
| 3.5.1 | Constructor & Destructor Documentation | 19 |
| 3.5.1.1 | LecteurCarte() | 19 |
| 3.5.2 | Member Function Documentation | 19 |
| 3.5.2.1 | _lire_carte_ajouter() | 19 |
| 3.5.2.2 | _lire_carte_supprimer() | 19 |
| 3.5.2.3 | connect_gene() | 20 |
| 3.5.2.4 | lecteurcarte_lire_carte() | 20 |
| 3.5.2.5 | lecteurcarte_relire_carte() | 20 |
| 3.5.2.6 | lire_carte() | 21 |
| 3.5.3 | Member Data Documentation | 21 |
| 3.5.3.1 | depart_timer | 21 |
| 3.5.3.2 | io | 21 |
| 3.5.3.3 | num_client | 21 |
| 3.6 | Prise Class Reference | 22 |
| 3.6.1 | Constructor & Destructor Documentation | 22 |
| 3.6.1.1 | Prise() | 22 |
| 3.6.2 | Member Function Documentation | 23 |
| 3.6.2.1 | attente_pdebranche() | 23 |
| 3.6.2.2 | deverouiller_trappe() | 23 |
| 3.6.2.3 | set_prise() | 23 |
| 3.6.2.4 | verouiller_trappe() | 24 |
| 3.7 | Timer Class Reference | 24 |
| 3.7.1 | Constructor & Destructor Documentation | 24 |
| 3.7.1.1 | Timer() | 24 |
| 3.7.2 | Member Function Documentation | 24 |

| | |
|--|-----------|
| 3.7.2.1 raz() | 25 |
| 3.7.2.2 valeur() | 25 |
| 3.8 Voyants Class Reference | 25 |
| 3.8.1 Constructor & Destructor Documentation | 26 |
| 3.8.1.1 Voyants() | 26 |
| 3.8.2 Member Function Documentation | 26 |
| 3.8.2.1 set_charge() | 26 |
| 3.8.2.2 set_dispo() | 26 |
| 3.8.3 Member Data Documentation | 28 |
| 3.8.3.1 depart_timer | 28 |
| 3.8.3.2 io | 28 |
| 3.8.3.3 num_client | 28 |
| 4 File Documentation | 29 |
| 4.1 base_donnee.h File Reference | 29 |
| 4.1.1 Detailed Description | 29 |
| 4.2 borne.h File Reference | 29 |
| 4.2.1 Detailed Description | 30 |
| 4.3 boutons.h File Reference | 30 |
| 4.3.1 Detailed Description | 30 |
| 4.4 lecteurcarte.h File Reference | 30 |
| 4.4.1 Detailed Description | 31 |
| 4.5 prise.h File Reference | 31 |
| 4.5.1 Detailed Description | 31 |
| 4.6 timer.h File Reference | 31 |
| 4.6.1 Detailed Description | 32 |
| 4.7 voyants.h File Reference | 32 |
| 4.7.1 Detailed Description | 32 |
| Index | 33 |

Chapter 1

Class Index

1.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

| | |
|-------------------------------|----|
| BaseDonnee | 5 |
| Borne | 11 |
| Boutons | 11 |
| Generateur | 13 |
| LecteurCarte | 18 |
| Prise | 22 |
| Timer | 24 |
| Voyants | 25 |

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

| | |
|---|----|
| base_donnee.h | |
| Gestion de la base de données clients | 29 |
| base_donneeold.h | ?? |
| borne.h | |
| Point d'entrée du programme C++ et de l'IHM | 29 |
| boutons.h | |
| Classe permettant à l'utilisateur d'interagir avec les boutons de l'IHM | 30 |
| generateur_save.h | ?? |
| lecteurcarte.h | |
| Classe permettant de gérer le lecteur de carte qui permet au client d'insérer sa carte abonné . | 30 |
| prise.h | |
| Classe permettant de gérer les actions sur la prise électrique du véhicule | 31 |
| timer.h | |
| Classe permettant de gérer les temporisations présentent dans le cahier des charges | 31 |
| voyants.h | |
| Classe permettant de gérer les voyants présents l'IHM de la borne | 32 |

Chapter 3

Class Documentation

3.1 BaseDonnee Class Reference

Public Member Functions

- **BaseDonnee** ()
Constructeur de la classe.
- int **authentifier** (unsigned short int num_carte)
Authentifier un client.
- int **re_authentifier** (unsigned short int num_carte)
Ré-Authentifier un client.
- void **ajouter_client** (unsigned short int num_carte)
Ajouter un client.
- void **supprimer_client** (unsigned short int num_carte)
Supprimer un client.
- void **init_base_client** ()
Initialiser la base de donnée.
- void **afficher_liste_client** ()
Afficher la liste client.
- void **ecriture_clients_fichier** ()
Ecrire la liste des clients.
- **BaseDonnee** ()
Constructeur de la classe.
- int **authentifier** (unsigned short int num_carte)
Authentifier un client.
- int **re_authentifier** (unsigned short int num_carte)
Ré-Authentifier un client.
- void **ajouter_client** (unsigned short int num_carte)
Ajouter un client.
- void **supprimer_client** (unsigned short int num_carte)
Supprimer un client.
- void **init_base_client** ()
Initialiser la base de donnée.
- void **afficher_liste_client** ()
Afficher la liste client.

Public Attributes

- `std::vector< int > base_donnees`
- `int i`
- `int dernier_client`
- `int nb_clients`
- `int base_donnees [100]`

3.1.1 Constructor & Destructor Documentation

3.1.1.1 BaseDonnee() [1/2]

```
BaseDonnee::BaseDonnee ( )
```

Constructeur de la classe.

Constructeur sans paramètre.

3.1.1.2 BaseDonnee() [2/2]

```
BaseDonnee::BaseDonnee ( )
```

Constructeur de la classe.

Constructeur sans paramètre.

3.1.2 Member Function Documentation

3.1.2.1 afficher_liste_client() [1/2]

```
void BaseDonnee::afficher_liste_client ( )
```

Afficher la liste client.

Methode qui permet d'afficher la liste des clients appartenants à la base de donnée.

3.1.2.2 afficher_liste_client() [2/2]

```
void BaseDonnee::afficher_liste_client ( )
```

Afficher la liste client.

Methode qui permet d'afficher la liste des clients appartenants à la base de donnée.

3.1.2.3 ajouter_client() [1/2]

```
void BaseDonnee::ajouter_client (
    unsigned short int num_carte )
```

Ajouter un client.

Methode qui permet d'ajouter un client à la base de données.

Parameters

| | |
|------------------|---|
| <i>num_carte</i> | : le numéro de la carte qui sera attribuée au client. |
|------------------|---|

3.1.2.4 ajouter_client() [2/2]

```
void BaseDonnee::ajouter_client (
    unsigned short int num_carte )
```

Ajouter un client.

Methode qui permet d'ajouter un client à la base de données.

Parameters

| | |
|------------------|---|
| <i>num_carte</i> | : le numéro de la carte qui sera attribuée au client. |
|------------------|---|

3.1.2.5 authentifier() [1/2]

```
int BaseDonnee::authentifier (
    unsigned short int num_carte )
```

Authentifier un client.

Methode qui permet d'authentifier un client appartenant à la base de données.

Parameters

| | |
|------------------|---|
| <i>num_carte</i> | : le numéro de la carte que possède le client |
|------------------|---|

Returns

1 si l'authentification réussit, 0 si elle échoue

3.1.2.6 authentifier() [2/2]

```
int BaseDonnee::authentifier (
    unsigned short int num_carte )
```

Authentifier un client.

Methode qui permet d'authentifier un client appartenant à la base de données.

Parameters

| | |
|------------------|---|
| <i>num_carte</i> | : le numéro de la carte que possède le client |
|------------------|---|

Returns

1 si l'authentification réussit, 0 si elle échoue

3.1.2.7 ecriture_clients_fichier()

```
void BaseDonnee::ecriture_clients_fichier ( )
```

Ecrire la liste des clients.

Methode qui permet d'écrire la liste des clients dans un fichier .TXT

3.1.2.8 init_base_client() [1/2]

```
void BaseDonnee::init_base_client ( )
```

Initialiser la base de donnée.

Methode qui permet d'initialiser la base de donnée au démarrage de l'IHM.

3.1.2.9 init_base_client() [2/2]

```
void BaseDonnee::init_base_client ( )
```

Initialiser la base de donnée.

Methode qui permet d'initialiser la base de donnée au démarrage de l'IHM.

3.1.2.10 re_authentifier() [1/2]

```
int BaseDonnee::re_authentifier (
    unsigned short int num_carte )
```

Ré-Authentifier un client.

Methode qui permet de ré-authentifier pour permettre la reprise du véhicule.

Parameters

| | |
|------------------|---|
| <i>num_carte</i> | : le numéro de la carte que possède le client |
|------------------|---|

Returns

1 si l'authentification réussit, 0 si elle échoue

3.1.2.11 re_authentifier() [2/2]

```
int BaseDonnee::re_authentifier (
    unsigned short int num_carte )
```

Ré-Authentifier un client.

Methode qui permet de ré-authentifier pour permettre la reprise du véhicule.

Parameters

| | |
|------------------|---|
| <i>num_carte</i> | : le numéro de la carte que possède le client |
|------------------|---|

Returns

1 si l'authentification réussit, 0 si elle échoue

3.1.2.12 supprimer_client() [1/2]

```
void BaseDonnee::supprimer_client (
    unsigned short int num_carte )
```

Supprimer un client.

Methode qui permet de supprimer un client de la base de données.

3.1.2.13 supprimer_client() [2/2]

```
void BaseDonnee::supprimer_client (
    unsigned short int num_carte )
```

Supprimer un client.

Methode qui permet de supprimer un client de la base de données.

3.1.3 Member Data Documentation**3.1.3.1 base_donnes [1/2]**

```
std::vector<int> BaseDonnee::base_donnes
```

Parameters

| | |
|---------------------|---|
| <i>base_donnees</i> | : Variable de type entier. Tableau de données contenant tous les clients enregistrés. |
|---------------------|---|

3.1.3.2 base_donnees [2/2]

```
int BaseDonnee::base_donnees[100]
```

Parameters

| | |
|---------------------|---|
| <i>base_donnees</i> | : Variable de type entier. Tableau de données contenant tous les clients enregistrés. |
|---------------------|---|

3.1.3.3 dernier_client

```
int BaseDonnee::dernier_client
```

Parameters

| | |
|-----------------------|--|
| <i>dernier_client</i> | : Entier permettant de mémoriser le dernier client ayant utilisé le système de recharge. |
|-----------------------|--|

3.1.3.4 i

```
int BaseDonnee::i
```

Parameters

| | |
|----------|--|
| <i>i</i> | : Entier servant de variable d'incrémementation. |
|----------|--|

3.1.3.5 nb_clients

```
int BaseDonnee::nb_clients
```

Parameters

| | |
|------------------|---|
| <i>nb_client</i> | : Entier permettant de mémoriser le nombre totale de clients enregistrés. |
|------------------|---|

The documentation for this class was generated from the following files:

- **base_donnee.h**
- base_donneeold.h
- base_donnee.cpp
- base_donneeold.cpp

3.2 Borne Class Reference

Public Member Functions

- **Borne ()**
Constructeur de la classe.

3.2.1 Constructor & Destructor Documentation

3.2.1.1 Borne()

```
Borne::Borne ( )
```

Constructeur de la classe.

Constructeur sans paramètre.

The documentation for this class was generated from the following file:

- **borne.h**

3.3 Boutons Class Reference

Public Member Functions

- **Boutons ()**
Constructeur de classe.
- int **charge ()**
Appuie sur bouton charge.
- int **stop ()**
Appuie sur bouton stop.

Public Attributes

- entrees * **io**
- int **shmid**
- int **depart_timer**

3.3.1 Constructor & Destructor Documentation

3.3.1.1 Boutons()

```
Boutons::Boutons ( )
```

Constructeur de classe.

Constructeur sans paramètre.

3.3.2 Member Function Documentation

3.3.2.1 charge()

```
int Boutons::charge ( )
```

Appuie sur bouton charge.

Methode qui permet de savoir si l'utilisateur appuie sur le bouton charge. (L'utilisateur dispose de 1 min pour appuyer sur le bouton).

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

Returns

Retourne 1 si le bouton est actionné, 0 sinon.

3.3.2.2 stop()

```
int Boutons::stop ( )
```

Appuie sur bouton stop.

Methode qui permet de savoir si l'utilisateur appuie sur le bouton stop.

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

Returns

Retourne 1 si le bouton est actionné, 0 sinon.

3.3.3 Member Data Documentation

3.3.3.1 depart_timer

```
int Boutons::depart_timer
```

Parameters

| | |
|---------------------|---|
| <i>depart_timer</i> | : Entier permettant utilise pour lancer le timer. |
|---------------------|---|

3.3.3.2 io

```
entrees* Boutons::io
```

Parameters

| | |
|------------|----------------------------|
| <i>*io</i> | : Pointeur de type entrees |
|------------|----------------------------|

The documentation for this class was generated from the following files:

- **boutons.h**
- boutons.cpp

3.4 Générateur Class Reference

Public Member Functions

- **Générateur** (**LecteurCarte** &lecteur, **Voyants** &voyants, **Prise** &prise, **Boutons** &boutons)
Constructeur.
- void **generateursave_initialiser** ()
Initialisation du Générateur Save.
- void **charger_batterie** (int etat_initial)
Charger batterie.
- void **generer_PWM** (pwm val)
Générer PWM.
- float **tension** ()
Tension.

- void **charger** ()
Charger.
- void **ouvrir_AC** ()
Charger.
- void **fermer_AC** ()
fermer
- void **deconnecter** ()
Déconnecter.

Public Attributes

- entrees * **io**
- int **shmid**
- int **depart_timer**
- unsigned short int **num_client**

3.4.1 Constructor & Destructor Documentation

3.4.1.1 Generateur()

```
Generateur::Generateur (
    LecteurCarte & lecteur,
    Voyants & voyants,
    Prise & prise,
    Boutons & boutons )
```

Constructeur.

Constructeur de la classe **Generateur** (p. 13)

Parameters

| | |
|---------------------|---|
| <i>constructeur</i> | : lecteur (objet de type LecteurCarte (p. 18)), voyants (objet de type Voyants (p. 25)), prise (objet de type Prise (p. 22)) et boutons (objet de type Boutons (p. 11)) |
|---------------------|---|

3.4.2 Member Function Documentation

3.4.2.1 charger()

```
void Generateur::charger ( )
```

Charger.

Methode qui permet de retourner la valeur de la tension

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

Returns

Retourne la valeur de la tension

3.4.2.2 charger_batterie()

```
void Generateur::charger_batterie (
    int etat_initial )
```

Charger batterie.

Méthode contenant une machine à états pour permettre le rechargement de la batterie

Parameters

| | |
|---------------------|--|
| <i>etat_initial</i> | : entier contenant l'état initial de la machine à état |
|---------------------|--|

3.4.2.3 deconnecter()

```
void Generateur::deconnecter ( )
```

Déconnecter.

Methode qui permet de deconnecter la voiture de la borne de recharge

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

Returns

sans paramètre

3.4.2.4 fermer_AC()

```
void Generateur::fermer_AC ( )
```

fermer

Methode qui permet fermer le contacteur AC

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

Returns

sans paramètre

3.4.2.5 generateursave_initialiser()

```
void Generateur::generateursave_initialiser ( )
```

Initialisation du Générateur Save.

Methode qui permet d'initialiser le générateur Save

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.4.2.6 generer_PWM()

```
void Generateur::generer_PWM (
    pwm val )
```

Générer PWM.

Méthode permettant de générer un signal PWM.

Parameters

| | |
|------------|------------------------|
| <i>val</i> | : variable de type pwm |
|------------|------------------------|

3.4.2.7 ouvrir_AC()

```
void Generateur::ouvrir_AC ( )
```

Charger.

Methode qui permet de retourner la valeur de la tension

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

Returns

Retourne la valeur de la tension

3.4.2.8 tension()

```
float Generateur::tension ( )
```

Tension.

Methode qui permet de retourner la valeur de la tension

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

Returns

Retourne la valeur de la tension

3.4.3 Member Data Documentation**3.4.3.1 depart_timer**

```
int Generateur::depart_timer
```

Parameters

| | |
|---------------------|---|
| <i>depart_timer</i> | : Entier permettant utilise pour lancer le timer. |
|---------------------|---|

3.4.3.2 io

```
entrees* Generateur::io
```

Parameters

| | |
|------------------|----------------------------|
| <code>*io</code> | : Pointeur de type entrees |
|------------------|----------------------------|

3.4.3.3 num_client

```
unsigned short int Generateur::num_client
```

Parameters

| | |
|-------------------------|---|
| <code>num_client</code> | : Entier non signé contenant le numéro du client. |
|-------------------------|---|

The documentation for this class was generated from the following files:

- generateur_save.h
- generateur_save.cpp

3.5 LecteurCarte Class Reference

Public Member Functions

- **LecteurCarte** (**Boutons** &bouton, **Voyants** &voyant, **BaseDonnee** &bd, **Prise** &p)
Constructeur.
- void **lecteurcarte_lire_carte** ()
Lire la carte d'un client.
- void **lecteurcarte_relire_carte** ()
Relire la carte d'un client.
- void **_lire_carte_ajouter** ()
Ajouter une carte.
- void **_lire_carte_supprimer** ()
Supprimer une carte.
- void **lire_carte** ()
Lire la carte d'un client.
- void **connect_gene** (**Generateur** *g)
Fonction de connection de classes.

Public Attributes

- entrees * **io**
- int **shmid**
- int **depart_timer**
- unsigned short int **num_client**

3.5.1 Constructor & Destructor Documentation

3.5.1.1 LecteurCarte()

```
LecteurCarte::LecteurCarte (
    Boutons & bouton,
    Voyants & voyant,
    BaseDonnee & bd,
    Prise & p )
```

Constructeur.

Constructeur de la classe **LecteurCarte** (p. 18)

Parameters

| | |
|---------------------|--|
| <i>constructeur</i> | : bouton (objet de type Boutons (p. 11)), voyants (objet de type Voyants (p. 25)), bd (objet de type BaseDonnee (p. 5)) et prise (objet de type Prise (p. 22)) |
|---------------------|--|

3.5.2 Member Function Documentation

3.5.2.1 _lire_carte_ajouter()

```
void LecteurCarte::_lire_carte_ajouter ( )
```

Ajouter une carte.

Methode qui permet d'ajouter la carte lu par le lecteur carte à la base de données clients.

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.5.2.2 _lire_carte_supprimer()

```
void LecteurCarte::_lire_carte_supprimer ( )
```

Supprimer une carte.

Methode qui permet de supprimer la carte lu par le lecteur carte à la base de données clients.

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.5.2.3 connect_gene()

```
void LecteurCarte::connect_gene (
    Generateur * g ) [inline]
```

Fonction de connection de classes.

Methode qui permet de connecter la classe **Generateur** (p. 13) et **LecteurCarte** (p. 18)

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.5.2.4 lecteurcarte_lire_carte()

```
void LecteurCarte::lecteurcarte_lire_carte ( )
```

Lire la carte d'un client.

Methode qui permet de lire le numéro de la carte d'un client.

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.5.2.5 lecteurcarte_relire_carte()

```
void LecteurCarte::lecteurcarte_relire_carte ( )
```

Relire la carte d'un client.

Methode qui permet de relire le numéro de la carte d'un client.

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.5.2.6 lire_carte()

```
void LecteurCarte::lire_carte ( )
```

Lire la carte d'un client.

Methode qui permet de lire le numéro de la carte d'un client.

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.5.3 Member Data Documentation

3.5.3.1 depart_timer

```
int LecteurCarte::depart_timer
```

Parameters

| | |
|---------------------|---|
| <i>depart_timer</i> | : Variable de type entier utile pour pouvoir lancer un timer. |
|---------------------|---|

3.5.3.2 io

```
entrees* LecteurCarte::io
```

Parameters

| | |
|------------|----------------------------|
| <i>*io</i> | : Pointeur de type entrees |
|------------|----------------------------|

3.5.3.3 num_client

```
unsigned short int LecteurCarte::num_client
```

Parameters

| | |
|-------------------|--|
| <i>num_client</i> | : Variable de type entier non signé utile pour mémoriser le numéro de la carte que le client à inséré. |
|-------------------|--|

The documentation for this class was generated from the following files:

- `lecteurcarte.h`
- `lecteurcarte.cpp`

3.6 Prise Class Reference

Public Member Functions

- **Prise** (**Voyants** &voyant)
Constructeur.
- void **deverrouiller_trappe** ()
Déverrouiller la trappe de la prise.
- void **verrouiller_trappe** ()
Verrouiller la trappe de la prise.
- void **set_prise** (led val)
*Set **Prise** (p. 22) (à compléter)*
- void **attente_pdebranche** ()
*Attente **Prise** (p. 22) débranchée.*

Public Attributes

- **Voyants** & voyant
- entrees * io
- int **depart_timer**
- int **shmid**

3.6.1 Constructor & Destructor Documentation

3.6.1.1 Prise()

```
Prise::Prise (
    Voyants & voyant )
```

Constructeur.

Constructeur de la classe **Prise** (p. 22)

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | parametre |
|-------------|-----------|

3.6.2 Member Function Documentation

3.6.2.1 attente_pdebranche()

```
void Prise::attente_pdebranche ( )
```

Attente **Prise** (p. 22) débranchée.

Methode qui permet d'attentre que l'utilisateur débranche la prise

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.6.2.2 deverrouiller_trappe()

```
void Prise::deverrouiller_trappe ( )
```

Déverrouiller la trappe de la prise.

Methode qui permet de déverrouiller la trappe de la prise.

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.6.2.3 set_prise()

```
void Prise::set_prise (
    led val )
```

Set **Prise** (p. 22) (à completer)

Methode qui permet de ?????

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.6.2.4 verouiller_trappe()

```
void Prise::verouiller_trappe ( )
```

Verrouiller la trappe de la prise.

Methode qui permet de verrouiller la trappe de la prise.

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

The documentation for this class was generated from the following files:

- **prise.h**
- prise.cpp

3.7 Timer Class Reference

Public Member Functions

- **Timer ()**
*Constructeur de la classe **Timer** (p. 24).*
- void **raz ()**
Remise à zéro du timer.
- int **valeur ()**
Valeur timer.

3.7.1 Constructor & Destructor Documentation

3.7.1.1 Timer()

```
Timer::Timer ( )
```

Constructeur de la classe **Timer** (p. 24).

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.7.2 Member Function Documentation

3.7.2.1 raz()

```
void Timer::raz ( )
```

Remise à zéro du timer.

Methode qui permet de remettre à zéro le timer.

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.7.2.2 valeur()

```
int Timer::valeur ( )
```

Valeur timer.

Methode qui permet de connaitre la valeur renvoyée par le timer.

Returns

Retourne un entier avec la valeur du timer.

The documentation for this class was generated from the following files:

- **timer.h**
- timer.cpp

3.8 Voyants Class Reference

Public Member Functions

- **Voyants ()**
*Constructeur de la classe **Voyants** (p. 25).*
- void **set_charge** (led val)
Set charge (à compléter)
- void **set_dispo** (led val)
Set dispo (à compléter)
- void **blink_charge** (led val)
- void **set_default** (led val)
- void **set_defaultOFF** (led val)

Public Attributes

- entrees * **io**
- int **shmid**
- int **depart_timer**
- unsigned short int **num_client**

3.8.1 Constructor & Destructor Documentation

3.8.1.1 Voyants()

```
Voyants::Voyants ( )
```

Constructeur de la classe **Voyants** (p. 25).

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.8.2 Member Function Documentation

3.8.2.1 set_charge()

```
void Voyants::set_charge (
    led val )
```

Set charge (à completer)

Methode qui permet de ?????

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.8.2.2 set_dispo()

```
void Voyants::set_dispo (
    led val )
```


Set dispo (à compléter)

Methode qui permet de ?????

Parameters

| | |
|-------------|-----------|
| <i>sans</i> | paramètre |
|-------------|-----------|

3.8.3 Member Data Documentation

3.8.3.1 depart_timer

```
int Voyants::depart_timer
```

Parameters

| | |
|---------------------|---|
| <i>depart_timer</i> | : Variable de type entier utile pour pouvoir lancer un timer. |
|---------------------|---|

3.8.3.2 io

```
entrees* Voyants::io
```

Parameters

| | |
|------------|----------------------------|
| <i>*io</i> | : Pointeur de type entrees |
|------------|----------------------------|

3.8.3.3 num_client

```
unsigned short int Voyants::num_client
```

Parameters

| | |
|-------------------|--|
| <i>num_client</i> | : Variable de type entier non signé utile pour mémoriser le numéro de la carte que le client à inséré. |
|-------------------|--|

The documentation for this class was generated from the following files:

- **voyants.h**
- voyants.cpp

Chapter 4

File Documentation

4.1 base_donnee.h File Reference

Gestion de la base de données clients.

```
#include <iostream>
#include <vector>
#include <fstream>
```

Classes

- class **BaseDonnee**

4.1.1 Detailed Description

Gestion de la base de données clients.

Author

JENN ALET et QUIVET

4.2 borne.h File Reference

Point d'entrée du programme C++ et de l'IHM.

```
#include <lcarte.h>
#include "memoire_borne.h"
#include "donnees_borne.h"
#include "lecteurcarte.h"
#include "voyants.h"
#include "boutons.h"
#include "prise.h"
#include "timer.h"
#include "generateur_save.h"
#include "base_donnee.h"
```

Classes

- class **Borne**

4.2.1 Detailed Description

Point d'entrée du programme C++ et de l'IHM.

Author

JENN ALET et QUIVET

4.3 boutons.h File Reference

Classe permettant à l'utilisateur d'interagir avec les boutons de l'IHM.

```
#include <iostream>
#include "donnees_borne.h"
#include "memoire_borne.h"
#include "timer.h"
```

Classes

- class **Boutons**

4.3.1 Detailed Description

Classe permettant à l'utilisateur d'interagir avec les boutons de l'IHM.

Author

JENN ALET et QUIVET

4.4 lecteurcarte.h File Reference

Classe permettant de gérer le lecteur de carte qui permet au client d'insérer sa carte abonné.

```
#include <iostream>
#include <lcarte.h>
#include <unistd.h>
#include "donnees_borne.h"
#include "memoire_borne.h"
#include "base_donnee.h"
#include "voyants.h"
#include "boutons.h"
#include "generateur_save.h"
#include "prise.h"
```

Classes

- class **LecteurCarte**

4.4.1 Detailed Description

Classe permettant de gérer le lecteur de carte qui permet au client d'insérer sa carte abonné.

Author

JENN ALET et QUIVET

4.5 prise.h File Reference

Classe permettant de gérer les actions sur la prise électrique du véhicule.

```
#include <iostream>
#include "donnees_borne.h"
#include "memoire_borne.h"
#include "voyants.h"
```

Classes

- class **Prise**

4.5.1 Detailed Description

Classe permettant de gérer les actions sur la prise électrique du véhicule.

Author

JENN ALET et QUIVET

4.6 timer.h File Reference

Classe permettant de gérer les temporisations présentent dans le cahier des charges.

```
#include <iostream>
#include "donnees_borne.h"
#include "memoire_borne.h"
```

Classes

- class **Timer**

4.6.1 Detailed Description

Classe permettant de gérer les temporisations présentent dans le cahier des charges.

Author

JENN ALET et QUIVET

4.7 voyants.h File Reference

Classe permettant de gérer les voyants présents l'IHM de la borne.

```
#include <iostream>
#include <lcarte.h>
#include <unistd.h>
#include "donnees_borne.h"
#include "memoire_borne.h"
```

Classes

- class **Voyants**

4.7.1 Detailed Description

Classe permettant de gérer les voyants présents l'IHM de la borne.

Author

JENN ALET et QUIVET

Index

- `_lire_carte_ajouter`
 - LecteurCarte, 19
 - `_lire_carte_supprimer`
 - LecteurCarte, 19
- `afficher_liste_client`
 - BaseDonnee, 6
- `ajouter_client`
 - BaseDonnee, 6, 7
- `attente_pdebranche`
 - Prise, 23
- `authentifier`
 - BaseDonnee, 7
- `base_donnee.h`, 29
- `base_donnees`
 - BaseDonnee, 9, 10
- BaseDonnee, 5
 - `afficher_liste_client`, 6
 - `ajouter_client`, 6, 7
 - `authentifier`, 7
 - `base_donnees`, 9, 10
 - BaseDonnee, 6
 - `dernier_client`, 10
 - `ecriture_clients_fichier`, 8
 - i, 10
 - `init_base_client`, 8
 - `nb_clients`, 10
 - `re_authentifier`, 8, 9
 - `supprimer_client`, 9
- Borne, 11
 - Borne, 11
- `borne.h`, 29
- Boutons, 11
 - Boutons, 12
 - `charge`, 12
 - `depart_timer`, 13
 - io, 13
 - `stop`, 12
- `boutons.h`, 30
- `charge`
 - Boutons, 12
- `charger`
 - Generateur, 14
- `charger_batterie`
 - Generateur, 15
- `connect_gene`
 - LecteurCarte, 20
- `deconnecter`
 - Generateur, 15
- `depart_timer`
 - Boutons, 13
 - Generateur, 17
 - LecteurCarte, 21
 - Voyants, 28
- `dernier_client`
 - BaseDonnee, 10
- `deverouiller_trappe`
 - Prise, 23
- `ecriture_clients_fichier`
 - BaseDonnee, 8
- `fermer_AC`
 - Generateur, 15
- Generateur, 13
 - `charger`, 14
 - `charger_batterie`, 15
 - `deconnecter`, 15
 - `depart_timer`, 17
 - `fermer_AC`, 15
 - Generateur, 14
 - `generateursave_initialiser`, 16
 - `generer_PWM`, 16
 - io, 17
 - `num_client`, 18
 - `ouvrir_AC`, 16
 - `tension`, 17
- `generateursave_initialiser`
 - Generateur, 16
- `generer_PWM`
 - Generateur, 16
- i
 - BaseDonnee, 10
- `init_base_client`
 - BaseDonnee, 8
- io
 - Boutons, 13
 - Generateur, 17
 - LecteurCarte, 21
 - Voyants, 28
- LecteurCarte, 18
 - `_lire_carte_ajouter`, 19
 - `_lire_carte_supprimer`, 19
 - `connect_gene`, 20
 - `depart_timer`, 21
 - io, 21

- LecteurCarte, 19
- lecteurcarte_lire_carte, 20
- lecteurcarte_relire_carte, 20
- lire_carte, 21
- num_client, 21
- lecteurcarte.h, 30
- lecteurcarte_lire_carte
 - LecteurCarte, 20
- lecteurcarte_relire_carte
 - LecteurCarte, 20
- lire_carte
 - LecteurCarte, 21
- nb_clients
 - BaseDonnee, 10
- num_client
 - Generateur, 18
 - LecteurCarte, 21
 - Voyants, 28
- ouvrir_AC
 - Generateur, 16
- Prise, 22
 - attente_pdebranche, 23
 - deverouiller_trappe, 23
 - Prise, 22
 - set_prise, 23
 - verouiller_trappe, 23
- prise.h, 31
- raz
 - Timer, 24
- re_authentifier
 - BaseDonnee, 8, 9
- set_charge
 - Voyants, 26
- set_dispo
 - Voyants, 26
- set_prise
 - Prise, 23
- stop
 - Boutons, 12
- supprimer_client
 - BaseDonnee, 9
- tension
 - Generateur, 17
- Timer, 24
 - raz, 24
 - Timer, 24
 - valeur, 25
- timer.h, 31
- valeur
 - Timer, 25
- verouiller_trappe
 - Prise, 23
- Voyants, 25
- depart_timer, 28
- io, 28
- num_client, 28
- set_charge, 26
- set_dispo, 26
- Voyants, 26
- voyants.h, 32