

SAPEICore

Generado por Doxygen 1.9.4

| | |
|---|-----------|
| 1 Índice jerárquico | 1 |
| 1.1 Jerarquía de la clase | 1 |
| 2 Índice de clases | 3 |
| 2.1 Lista de clases | 3 |
| 3 Índice de archivos | 5 |
| 3.1 Lista de archivos | 5 |
| 4 Documentación de las clases | 7 |
| 4.1 Referencia de la Clase Client | 7 |
| 4.1.1 Descripción detallada | 8 |
| 4.1.2 Documentación de los datos miembro | 8 |
| 4.1.2.1 balance | 8 |
| 4.1.2.2 id | 8 |
| 4.1.2.3 vehicle | 9 |
| 4.2 Referencia de la Clase Person | 9 |
| 4.3 Referencia de la Clase Vehicle | 10 |
| 5 Documentación de archivos | 11 |
| 5.1 /home/corte/Documents/SAPEI/SAPEICore/Client.h | 11 |
| 5.2 /home/corte/Documents/SAPEI/SAPEICore/Person.h | 11 |
| 5.3 /home/corte/Documents/SAPEI/SAPEICore/Vehicle.h | 12 |

Chapter 1

Índice jerárquico

1.1 Jerarquía de la clase

Esta lista de herencias esta ordenada aproximadamente por orden alfabético:

| | |
|-------------------|----|
| Person | 9 |
| Client | 7 |
| Vehicle | 10 |

Chapter 2

Índice de clases

2.1 Lista de clases

Lista de las clases, estructuras, uniones e interfaces con una breve descripción:

| | | |
|-------------------------|--|--------------------|
| Client | Clase para administrar los datos de un cliente | 7 |
| Person | | 9 |
| Vehicle | | 10 |

Chapter 3

Indice de archivos

3.1 Lista de archivos

Lista de todos los archivos documentados y con descripciones breves:

| | |
|--|----|
| /home/corte/Documents/SAPEI/SAPEICore/ Client.h | ?? |
| /home/corte/Documents/SAPEI/SAPEICore/ Person.h | ?? |
| /home/corte/Documents/SAPEI/SAPEICore/ Vehicle.h | ?? |

Chapter 4

Documentación de las clases

4.1 Referencia de la Clase Client

Clase para administrar los datos de un cliente.

```
#include <Client.h>
```

Diagrama de herencias de Client

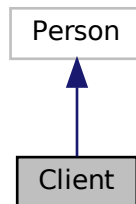
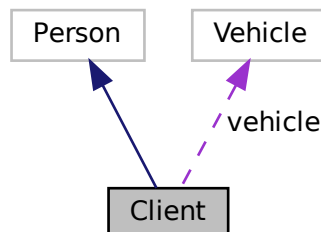


Diagrama de colaboración para Client:



Métodos públicos

- **Client** (int [id](#), const std::string &name, int age=0, const std::string &address="", const std::string &email="", const std::string &phone="", const std::string &license="", const std::string &type="", const std::string &color="", const std::string &brand="", const std::string &model="")
- int **getId** () const
- int **getBalance** () const
- void **setId** (int [id](#))
- void **setVehicle** (const [Vehicle](#) &[vehicle](#))
- void **setBalance** (float [balance](#))

Atributos privados

- int [id](#)
- [Vehicle](#) [vehicle](#)
- float [balance](#)

Amigas

- std::ostream & **operator**<< (std::ostream &os, const [Client](#) &[Client](#))

4.1.1 Descripción detallada

Clase para administrar los datos de un cliente.

4.1.2 Documentación de los datos miembro

4.1.2.1 balance

```
float Client::balance [private]
```

Lleva el saldo del cliente, en pesos argentinos

4.1.2.2 id

```
int Client::id [private]
```

ID relacionado a la tarjeta RFID unica de cada cliente

4.1.2.3 vehicle

```
Vehicle Client::vehicle [private]
```

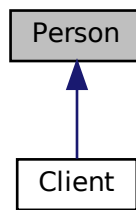
Objeto que almacena los datos del vehiculo

La documentación para esta clase fue generada a partir de los siguientes ficheros:

- /home/corte/Documents/SAPEI/SAPEICore/Client.h
- /home/corte/Documents/SAPEI/SAPEICore/Client.cpp

4.2 Referencia de la Clase Person

Diagrama de herencias de Person



Métodos públicos

- **Person** (const std::string &name="", int age=0, const std::string &address="", const std::string &email="", const std::string &phone="")
- std::string **getName** () const
- int **getAge** () const
- std::string **getAddress** () const

Atributos privados

- std::string **name**
- int **age**
- std::string **address**
- std::string **email**
- std::string **phone**

Amigas

- `std::ostream & operator<< (std::ostream &os, const Person &person)`

La documentación para esta clase fue generada a partir de los siguientes ficheros:

- `/home/corte/Documents/SAPEI/SAPEICore/Person.h`
- `/home/corte/Documents/SAPEI/SAPEICore/Person.cpp`

4.3 Referencia de la Clase Vehicle

Métodos públicos

- **Vehicle** (const std::string &license="", const std::string &type="", const std::string &color="", const std::string &brand="", const std::string &model="")
- std::string **getLicensePlate** () const
- std::string **getType** () const
- std::string **getColor** () const
- std::string **getBrand** () const
- std::string **getModel** () const
- void **setLicensePlate** (const std::string &license)
- void **setType** (const std::string &type)
- void **setColor** (const std::string &color)
- void **setBrand** (const std::string &brand)
- void **setModel** (const std::string &model)

Atributos privados

- std::string **licensePlate**
- std::string **type**
- std::string **color**
- std::string **brand**
- std::string **model**

Amigas

- `std::ostream & operator<< (std::ostream &os, const Vehicle &vehicle)`

La documentación para esta clase fue generada a partir de los siguientes ficheros:

- `/home/corte/Documents/SAPEI/SAPEICore/Vehicle.h`
- `/home/corte/Documents/SAPEI/SAPEICore/Vehicle.cpp`

Chapter 5

Documentación de archivos

5.1 /home/corte/Documents/SAPEI/SAPEICore/Client.h

```
1 #ifndef CLIENT_H
2 #define CLIENT_H
3
4 #include "Person.h"
5 #include "Vehicle.h"
6
10 class Client : public Person {
11
12 friend std::ostream& operator<<(std::ostream& os, const Client& Client);
13
14 private:
15     int id;
16     Vehicle vehicle;
17     float balance;
18
19 public:
20     Client( int id, const std::string& name, int age = 0, const std::string& address = "", const
        std::string& email = "",
21             const std::string& phone = "", const std::string& license = "", const std::string& type = "",
22             const std::string& color = "", const std::string& brand = "", const std::string& model = "");
23
24     int getId() const;
25     int getBalance() const;
26
27     void setId(int id);
28     void setVehicle(const Vehicle& vehicle);
29     void setBalance(float balance);
30 };
31
32 #endif
```

5.2 /home/corte/Documents/SAPEI/SAPEICore/Person.h

```
1 #ifndef PERSON_H
2 #define PERSON_H
3
4 #include <iostream>
5 #include <string>
6
7 class Person {
8
9 friend std::ostream& operator<<(std::ostream& os, const Person& person);
10
11 private:
12     std::string name;
13     int age;
14     std::string address;
15     std::string email;
16     std::string phone;
17
18 public:
19     Person(const std::string& name = "", int age = 0, const std::string& address = "", const std::string&
        email = "", const std::string& phone = "");
20
21     std::string getName() const;
```

```
22     int getAge() const;
23     std::string getAddress() const;
24
25 };
26
27 #endif // PERSON_H
```

5.3 /home/corte/Documents/SAPEI/SAPEICore/Vehicle.h

```
1 #ifndef VEHICLE_H
2 #define VEHICLE_H
3
4 #include <iostream>
5 #include <string>
6
7 class Vehicle {
8
9     friend std::ostream& operator<<(std::ostream& os, const Vehicle& vehicle);
10
11 private:
12     std::string licensePlate;
13     std::string type;
14     std::string color;
15     std::string brand;
16     std::string model;
17
18 public:
19     Vehicle(const std::string& license = "", const std::string& type = "", const std::string& color = "",
20             const std::string& brand = "", const std::string& model = "");
21
22     std::string getLicensePlate() const;
23     std::string getType() const;
24     std::string getColor() const;
25     std::string getBrand() const;
26     std::string getModel() const;
27
28     void setLicensePlate(const std::string& license);
29     void setType(const std::string& type);
30     void setColor(const std::string& color);
31     void setBrand(const std::string& brand);
32     void setModel(const std::string& model);
33 };
34
35 #endif
```