

## Practice 4

Generated by Doxygen 1.8.17



<b>1 Hierarchical Index</b>	<b>1</b>
1.1 Class Hierarchy	1
<b>2 Class Index</b>	<b>3</b>
2.1 Class List	3
<b>3 File Index</b>	<b>5</b>
3.1 File List	5
<b>4 Class Documentation</b>	<b>7</b>
4.1 Admin Class Reference	7
4.1.1 Constructor & Destructor Documentation	8
4.1.1.1 Admin()	8
4.2 AirQuality Class Reference	8
4.2.1 Constructor & Destructor Documentation	9
4.2.1.1 AirQuality()	9
4.2.2 Member Function Documentation	9
4.2.2.1 requestData()	10
4.3 CameraBW Class Reference	10
4.3.1 Constructor & Destructor Documentation	11
4.3.1.1 CameraBW()	11
4.3.2 Member Function Documentation	11
4.3.2.1 requestData()	11
4.4 CameraRGB Class Reference	12
4.4.1 Constructor & Destructor Documentation	12
4.4.1.1 CameraRGB()	12
4.4.2 Member Function Documentation	13
4.4.2.1 requestData()	13
4.5 CLDashboard Class Reference	13
4.5.1 Constructor & Destructor Documentation	14
4.5.1.1 CLDashboard()	15
4.5.2 Member Function Documentation	15
4.5.2.1 changeMainMenu()	15
4.5.2.2 errorCommand()	15
4.5.2.3 helpCommand()	15
4.5.2.4 listSensor()	16
4.5.2.5 permissionError()	16
4.5.2.6 readCommand()	16
4.5.2.7 showMainMenu()	16
4.6 CLInterface Class Reference	17
4.6.1 Constructor & Destructor Documentation	17
4.6.1.1 CLInterface()	18
4.6.2 Member Function Documentation	18

4.6.2.1 login()	18
4.7 CLLoginInterface Class Reference	19
4.7.1 Member Function Documentation	20
4.7.1.1 askEmployeeNumber() [1/2]	20
4.7.1.2 askEmployeeNumber() [2/2]	20
4.7.1.3 askNIF() [1/2]	20
4.7.1.4 askNIF() [2/2]	21
4.7.1.5 showWelcomeMessage()	21
4.8 CLMenu Class Reference	21
4.8.1 Member Function Documentation	22
4.8.1.1 clearMenu()	22
4.8.1.2 show() [1/2]	22
4.8.1.3 show() [2/2]	23
4.9 CLMenuBar Class Reference	23
4.9.1 Member Function Documentation	24
4.9.1.1 show()	24
4.10 Dashboard Class Reference	25
4.10.1 Member Function Documentation	26
4.10.1.1 addNewSensor()	26
4.10.1.2 addToMainMenu() [1/2]	27
4.10.1.3 addToMainMenu() [2/2]	27
4.10.1.4 canExit()	27
4.10.1.5 changeCurrentSensorInfo()	28
4.10.1.6 changeInterface()	28
4.10.1.7 cleanSensor()	28
4.10.1.8 Create()	28
4.10.1.9 deleteSensor()	29
4.10.1.10 exit()	29
4.10.1.11 logout()	29
4.10.1.12 moveWindowMainMenu()	30
4.10.1.13 setUser()	30
4.11 Database Class Reference	30
4.12 Humidity Class Reference	31
4.12.1 Constructor & Destructor Documentation	31
4.12.1.1 Humidity()	31
4.12.2 Member Function Documentation	32
4.12.2.1 requestData()	32
4.13 Interface Class Reference	32
4.13.1 Member Function Documentation	33
4.13.1.1 Create()	33
4.13.1.2 loadMenu()	34
4.13.1.3 login()	34

---

4.14 LoginInterface Class Reference	35
4.14.1 Member Function Documentation	36
4.14.1.1 checkUser()	36
4.14.1.2 Create()	36
4.14.1.3 getUser()	36
4.15 Menu Class Reference	37
4.15.1 Member Function Documentation	37
4.15.1.1 Create()	37
4.16 MenuBar Class Reference	38
4.16.1 Member Function Documentation	39
4.16.1.1 Create()	39
4.16.1.2 setCurrentMenu()	39
4.16.1.3 setUsername()	39
4.17 Moisture Class Reference	40
4.17.1 Constructor & Destructor Documentation	40
4.17.1.1 Moisture()	41
4.17.2 Member Function Documentation	41
4.17.2.1 requestData()	41
4.18 Sensor Class Reference	41
4.18.1 Constructor & Destructor Documentation	42
4.18.1.1 Sensor()	43
4.18.2 Member Function Documentation	43
4.18.2.1 Create()	43
4.18.2.2 getArea()	43
4.18.2.3 getId()	44
4.18.2.4 getMagnitude()	44
4.18.2.5 getType()	44
4.18.2.6 getValPerMin()	44
4.18.2.7 isActive()	45
4.18.2.8 setActive()	45
4.18.2.9 setArea()	45
4.18.2.10 setMagnitude()	45
4.18.2.11 setValPerMin()	46
4.19 Thermometer Class Reference	46
4.19.1 Constructor & Destructor Documentation	47
4.19.1.1 Thermometer()	47
4.19.2 Member Function Documentation	47
4.19.2.1 requestData()	47
4.20 User Class Reference	48
4.20.1 Constructor & Destructor Documentation	49
4.20.1.1 User()	49
4.20.2 Member Function Documentation	49

---

4.20.2.1 addTimestamp()	49
4.20.2.2 getName()	49
4.20.2.3 getTimestamp()	50
4.20.2.4 hasAdminPermission()	50
4.20.2.5 isSameEmployeeNumber()	50
4.20.2.6 isSameNIF()	50
4.20.2.7 setEmployeeNumber()	51
4.20.2.8 setNIF()	51
<b>5 File Documentation</b>	<b>53</b>
5.1 include/Admin.h File Reference	53
5.1.1 Detailed Description	54
5.2 include/AirQuality.h File Reference	55
5.2.1 Detailed Description	55
5.3 include/CameraBW.h File Reference	56
5.3.1 Detailed Description	57
5.4 include/CameraRGB.h File Reference	57
5.4.1 Detailed Description	58
5.5 include/CLDashboard.h File Reference	59
5.5.1 Detailed Description	59
5.6 include/CLInterface.h File Reference	60
5.6.1 Detailed Description	61
5.7 include/CLIUtils.h File Reference	61
5.7.1 Detailed Description	62
5.7.2 Function Documentation	62
5.7.2.1 printCenter() [1/2]	62
5.7.2.2 printCenter() [2/2]	63
5.7.2.3 printCenterFromFile()	63
5.7.2.4 printColor()	63
5.7.2.5 setColor()	64
5.8 include/CLLoginInterface.h File Reference	64
5.8.1 Detailed Description	65
5.9 include/CLMenu.h File Reference	66
5.9.1 Detailed Description	66
5.10 include/CLMenuBar.h File Reference	67
5.10.1 Detailed Description	68
5.11 include/Dashboard.h File Reference	68
5.11.1 Detailed Description	69
5.12 include/Database.h File Reference	70
5.12.1 Detailed Description	71
5.13 include/Humidity.h File Reference	71
5.13.1 Detailed Description	72

5.14 include/Interface.h File Reference . . . . .	73
5.14.1 Detailed Description . . . . .	74
5.15 include/LoginInterface.h File Reference . . . . .	74
5.15.1 Detailed Description . . . . .	75
5.16 include/Menu.h File Reference . . . . .	76
5.16.1 Detailed Description . . . . .	76
5.17 include/MenuBar.h File Reference . . . . .	77
5.17.1 Detailed Description . . . . .	78
5.18 include/Moisture.h File Reference . . . . .	78
5.18.1 Detailed Description . . . . .	79
5.19 include/Sensor.h File Reference . . . . .	80
5.19.1 Detailed Description . . . . .	80
5.20 include/Thermometer.h File Reference . . . . .	81
5.20.1 Detailed Description . . . . .	81
5.21 include/User.h File Reference . . . . .	82
5.21.1 Detailed Description . . . . .	83
5.22 src/Admin.cpp File Reference . . . . .	83
5.22.1 Detailed Description . . . . .	84
5.23 src/AirQuality.cpp File Reference . . . . .	84
5.23.1 Detailed Description . . . . .	85
5.24 src/CameraBW.cpp File Reference . . . . .	85
5.24.1 Detailed Description . . . . .	86
5.25 src/CameraRGB.cpp File Reference . . . . .	86
5.25.1 Detailed Description . . . . .	87
5.26 src/CLDashboard.cpp File Reference . . . . .	87
5.26.1 Detailed Description . . . . .	87
5.27 src/CLInterface.cpp File Reference . . . . .	88
5.27.1 Detailed Description . . . . .	88
5.28 src/CLIUtils.cpp File Reference . . . . .	89
5.28.1 Detailed Description . . . . .	90
5.28.2 Function Documentation . . . . .	90
5.28.2.1 printCenter() [1/2] . . . . .	90
5.28.2.2 printCenter() [2/2] . . . . .	90
5.28.2.3 printCenterFromFile() . . . . .	91
5.28.2.4 printColor() . . . . .	91
5.28.2.5 setColor() . . . . .	91
5.29 src/CLLoginInterface.cpp File Reference . . . . .	92
5.29.1 Detailed Description . . . . .	92
5.30 src/CLMenu.cpp File Reference . . . . .	93
5.30.1 Detailed Description . . . . .	93
5.31 src/CLMenuBar.cpp File Reference . . . . .	94
5.31.1 Detailed Description . . . . .	94

---

5.32 src/Dashboard.cpp File Reference . . . . .	95
5.32.1 Detailed Description . . . . .	95
5.33 src/Database.cpp File Reference . . . . .	96
5.33.1 Detailed Description . . . . .	96
5.34 src/Humidity.cpp File Reference . . . . .	97
5.34.1 Detailed Description . . . . .	97
5.35 src/Interface.cpp File Reference . . . . .	98
5.35.1 Detailed Description . . . . .	98
5.36 src/LoginInterface.cpp File Reference . . . . .	99
5.36.1 Detailed Description . . . . .	99
5.37 src/main.cpp File Reference . . . . .	100
5.37.1 Detailed Description . . . . .	100
5.38 src/Menu.cpp File Reference . . . . .	101
5.38.1 Detailed Description . . . . .	101
5.39 src/MenuBar.cpp File Reference . . . . .	102
5.39.1 Detailed Description . . . . .	102
5.40 src/Moisture.cpp File Reference . . . . .	103
5.40.1 Detailed Description . . . . .	103
5.41 src/Sensor.cpp File Reference . . . . .	104
5.41.1 Detailed Description . . . . .	104
5.42 src/Thermometer.cpp File Reference . . . . .	105
5.42.1 Detailed Description . . . . .	105
5.43 src/User.cpp File Reference . . . . .	106
5.43.1 Detailed Description . . . . .	106
<b>Index</b>	<b>107</b>



# Chapter 1

## Hierarchical Index

### 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Dashboard . . . . .	25
CLDashboard . . . . .	13
Database . . . . .	30
Interface . . . . .	32
CLInterface . . . . .	17
LoginInterface . . . . .	35
CLLoginInterface . . . . .	19
Menu . . . . .	37
CLMenu . . . . .	21
MenuBar . . . . .	38
CLMenuBar . . . . .	23
Sensor . . . . .	41
AirQuality . . . . .	8
CameraBW . . . . .	10
CameraRGB . . . . .	12
Humidity . . . . .	31
Moisture . . . . .	40
Thermometer . . . . .	46
User . . . . .	48
Admin . . . . .	7



## Chapter 2

# Class Index

### 2.1 Class List

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

Admin	7
AirQuality	8
CameraBW	10
CameraRGB	12
CLDashboard	13
CLInterface	17
CLLoginInterface	19
CLMenu	21
CLMenuBar	23
Dashboard	25
Database	30
Humidity	31
Interface	32
LoginInterface	35
Menu	37
MenuBar	38
Moisture	40
Sensor	41
Thermometer	46
User	48



## Chapter 3

# File Index

### 3.1 File List

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

include/ <a href="#">Admin.h</a>	
Header file of <a href="#">Admin.cpp</a>	53
include/ <a href="#">AirQuality.h</a>	
Header file of <a href="#">AirQuality.cpp</a>	55
include/ <a href="#">CameraBW.h</a>	
Header file of <a href="#">CameraBW.cpp</a>	56
include/ <a href="#">CameraRGB.h</a>	
Header file of <a href="#">CameraRGB.cpp</a>	57
include/ <a href="#">CLDashboard.h</a>	
Header file of <a href="#">CLDashboard.cpp</a>	59
include/ <a href="#">CLInterface.h</a>	
Header file of <a href="#">CLInterface.cpp</a>	60
include/ <a href="#">CLIUtils.h</a>	
Header file of <a href="#">CLIUtils.cpp</a>	61
include/ <a href="#">CLLoginInterface.h</a>	
Header file of <a href="#">CLLoginInterface.cpp</a>	64
include/ <a href="#">CLMenu.h</a>	
Header file of <a href="#">CLMenu.cpp</a>	66
include/ <a href="#">CLMenuBar.h</a>	
Header file of <a href="#">CLMenuBar.cpp</a>	67
include/ <a href="#">Dashboard.h</a>	
Header file of <a href="#">Dashboard.cpp</a>	68
include/ <a href="#">Database.h</a>	
Header file of <a href="#">Database.cpp</a>	70
include/ <a href="#">Humidity.h</a>	
Header file of <a href="#">Humidity.cpp</a>	71
include/ <a href="#">Interface.h</a>	
Header file of <a href="#">Interface.cpp</a>	73
include/ <a href="#">LoginInterface.h</a>	
Header file of <a href="#">LoginInterface.cpp</a>	74
include/ <a href="#">Menu.h</a>	
Header file of <a href="#">Menu.cpp</a>	76
include/ <a href="#">MenuBar.h</a>	
Header file of <a href="#">MenuBar.cpp</a>	77
include/ <a href="#">Moisture.h</a>	
Header file of <a href="#">Moisture.cpp</a>	78

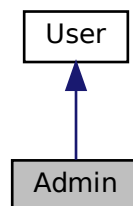
include/Sensor.h	
Header file of <a href="#">Sensor.cpp</a>	80
include/Thermometer.h	
Header file of <a href="#">Thermometer.cpp</a>	81
include/User.h	
Header file of <a href="#">User.cpp</a>	82
src/Admin.cpp	83
src/AirQuality.cpp	84
src/CameraBW.cpp	85
src/CameraRGB.cpp	86
src/CLDashboard.cpp	87
src/CLInterface.cpp	88
src/CLIUtils.cpp	89
src/CLLoginInterface.cpp	92
src/CLMenu.cpp	93
src/CLMenuBar.cpp	94
src/Dashboard.cpp	95
src/Database.cpp	96
src/Humidity.cpp	97
src/Interface.cpp	98
src/LoginInterface.cpp	99
src/main.cpp	100
src/Menu.cpp	101
src/MenuBar.cpp	102
src/Moisture.cpp	103
src/Sensor.cpp	104
src/Thermometer.cpp	105
src/User.cpp	106

## Chapter 4

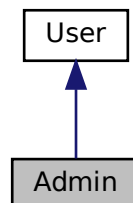
# Class Documentation

### 4.1 Admin Class Reference

Inheritance diagram for Admin:



Collaboration diagram for Admin:



### Public Member Functions

- [Admin](#) (std::string number="00000", std::string nif="000000000", std::string name="default")  
*Creates a new [Admin](#) object.*

## Additional Inherited Members

### 4.1.1 Constructor & Destructor Documentation

#### 4.1.1.1 Admin()

```
Admin::Admin (
    std::string employeeNumber = "00000",
    std::string NIF = "000000000",
    std::string name = "default" )
```

Creates a new [Admin](#) object.

#### Parameters

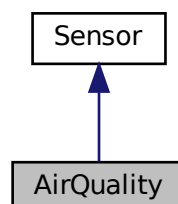
<i>employeeNumber</i>	
<i>NIF</i>	
<i>name</i>	

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

- include/[Admin.h](#)
- src/[Admin.cpp](#)

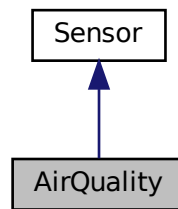
## 4.2 AirQuality Class Reference

Inheritance diagram for AirQuality:





Collaboration diagram for AirQuality:



## Public Member Functions

- `AirQuality` (bool active=false, int valPerMin=1)  
*Create a new Air Quality object.*
- `std::vector< int > requestData ()`  
*Request data from the air quality sensor.*

## Additional Inherited Members

### 4.2.1 Constructor & Destructor Documentation

#### 4.2.1.1 AirQuality()

```
AirQuality::AirQuality (
    bool active = false,
    int valPerMin = 1 )
```

Create a new Air Quality object.

#### Parameters

<i>active</i>	By default off
<i>valPerMin</i>	By default 1 val/min

### 4.2.2 Member Function Documentation

#### 4.2.2.1 requestData()

```
std::vector< int > AirQuality::requestData ( ) [virtual]
```

Request data from the air quality sensor.

##### Returns

std::vector<int> Data values

##### Note

This function is a placeholder, returns random values

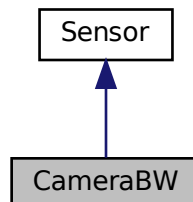
Implements [Sensor](#).

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

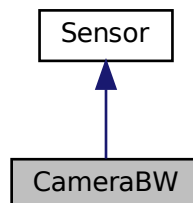
- include/[AirQuality.h](#)
- src/[AirQuality.cpp](#)

## 4.3 CameraBW Class Reference

Inheritance diagram for CameraBW:



Collaboration diagram for CameraBW:



## Public Member Functions

- [CameraBW](#) (bool active=false, int valPerMin=1)  
*Create a new BW camera object.*
- `std::vector< int >` [requestData](#) ()  
*Request data from the bw camera.*

## Additional Inherited Members

### 4.3.1 Constructor & Destructor Documentation

#### 4.3.1.1 CameraBW()

```
CameraBW::CameraBW (
    bool active = false,
    int valPerMin = 1 )
```

Create a new BW camera object.

##### Parameters

<i>active</i>	By default off
<i>valPerMin</i>	By default 1 val/min

### 4.3.2 Member Function Documentation

#### 4.3.2.1 requestData()

```
std::vector< int > CameraBW::requestData ( ) [virtual]
```

Request data from the bw camera.

##### Returns

`std::vector<int>` Data values

##### Note

This function is a placeholder, returns random values

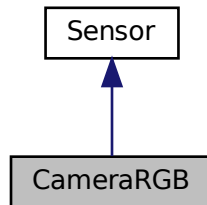
Implements [Sensor](#).

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

- include/[CameraBW.h](#)
- src/[CameraBW.cpp](#)

## 4.4 CameraRGB Class Reference

Inheritance diagram for CameraRGB:



Collaboration diagram for CameraRGB:



### Public Member Functions

- [CameraRGB](#) (bool active=false, int valPerMin=1)  
*Create a new RGB camera object.*
- std::vector< int > [requestData](#) ()  
*Request data from the rgb camera.*

### Additional Inherited Members

#### 4.4.1 Constructor & Destructor Documentation

##### 4.4.1.1 CameraRGB()

```
CameraRGB::CameraRGB (
    bool active = false,
    int valPerMin = 1 )
```

Create a new RGB camera object.

## Parameters

<i>active</i>	By default off
<i>valPerMin</i>	By default 1 val/min

## 4.4.2 Member Function Documentation

### 4.4.2.1 requestData()

```
std::vector< int > CameraRGB::requestData ( ) [virtual]
```

Request data from the rgb camera.

## Returns

std::vector<int> Data values

## Note

This function is a placeholder, returns random values

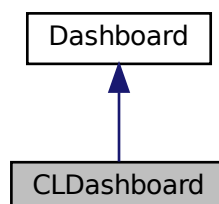
Implements [Sensor](#).

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

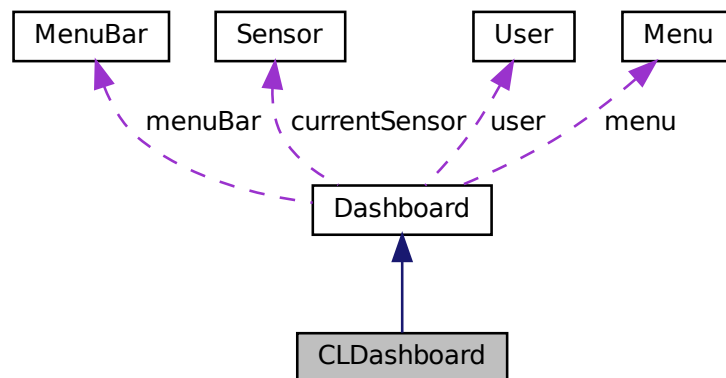
- include/[CameraRGB.h](#)
- src/[CameraRGB.cpp](#)

## 4.5 CLDashboard Class Reference

Inheritance diagram for CLDashboard:



Collaboration diagram for CLDashboard:



## Public Member Functions

- `CLDashboard ()`  
Create a new `Dashboard` for the command line.
- `void showMainMenu ()`  
Shows the default menu and starts the custom terminal.

## Protected Member Functions

- `void changeMainMenu (int n)`  
Change the page of the menu.
- `void readCommand ()`  
Reads and executes a command from the terminal.
- `void listSensor ()`  
Command that lists the sensors in the current menu page.
- `void helpCommand (std::string command="")`  
Displays the use of the command.
- `void errorCommand (std::string command="")`  
Shows error message because the command not being found.
- `void permissionError ()`  
Shows error message because the permission being denied to execute it.

## Additional Inherited Members

### 4.5.1 Constructor & Destructor Documentation

#### 4.5.1.1 CLDashboard()

```
CLDashboard::CLDashboard ( )
```

Create a new [Dashboard](#) for the command line.

### 4.5.2 Member Function Documentation

#### 4.5.2.1 changeMainMenu()

```
void CLDashboard::changeMainMenu (
    int n ) [protected]
```

Change the page of the menu.

##### Parameters

<i>n</i>	Number of pages to shift ( positive to the right )
----------	--

#### 4.5.2.2 errorCommand()

```
void CLDashboard::errorCommand (
    std::string command = "" ) [protected]
```

Shows error message because the command not being found.

##### Parameters

<i>command</i>	Command input
----------------	---------------

#### 4.5.2.3 helpCommand()

```
void CLDashboard::helpCommand (
    std::string command = "" ) [protected]
```

Displays the use of the command.

##### Parameters

<i>command</i>	Command to display info about
----------------	-------------------------------

#### 4.5.2.4 listSensor()

```
void CLDashboard::listSensor ( ) [protected]
```

Command that lists the sensors in the current menu page.

#### 4.5.2.5 permissionError()

```
void CLDashboard::permissionError ( ) [protected]
```

Shows error message because the permission being denied to execute it.

#### 4.5.2.6 readCommand()

```
void CLDashboard::readCommand ( ) [protected]
```

Reads and executes a command from the terminal.

#### 4.5.2.7 showMainMenu()

```
void CLDashboard::showMainMenu ( ) [virtual]
```

Shows the default menu and starts the custom terminal.

Implements [Dashboard](#).

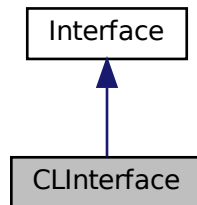
The documentation for this class was generated from the following files:

- [include/CLDashboard.h](#)
- [src/CLDashboard.cpp](#)

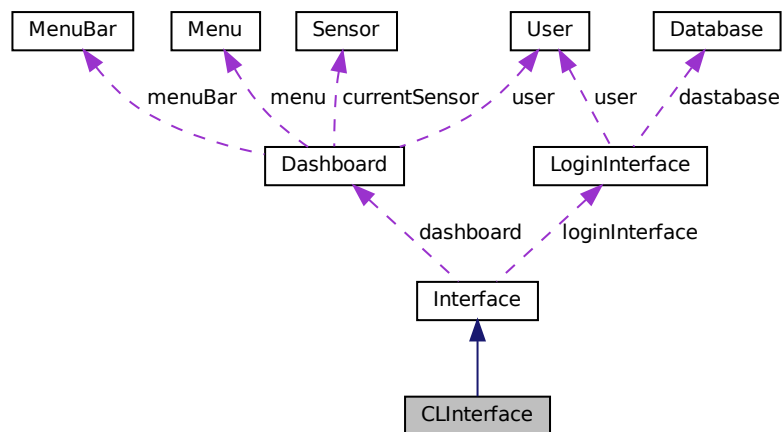


## 4.6 CLInterface Class Reference

Inheritance diagram for CLInterface:



Collaboration diagram for CLInterface:



### Public Member Functions

- `CLInterface` ()  
*Create a new [Interface](#) for the command line.*
- `void login` (int tries)  
*Controls the login interface for the command line.*
- `void loadMenu` ()

### Additional Inherited Members

#### 4.6.1 Constructor & Destructor Documentation

#### 4.6.1.1 CLInterface()

```
CLInterface::CLInterface ( )
```

Create a new [Interface](#) for the command line.

##### Note

Also gets the terminal size

### 4.6.2 Member Function Documentation

#### 4.6.2.1 login()

```
void CLInterface::login (
    int tries ) [virtual]
```

Controls the login interface for the command line.

##### Parameters

<i>tries</i>	Number of tries to login
--------------	--------------------------

##### Note

If exceeded the number of tries, then the program will exit

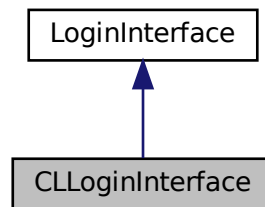
Implements [Interface](#).

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

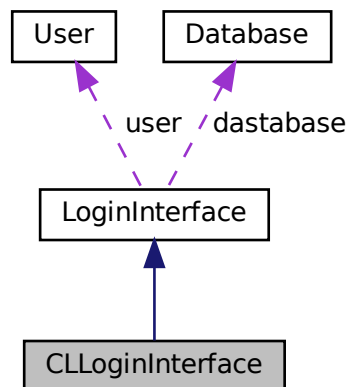
- [include/CLInterface.h](#)
- [src/CLInterface.cpp](#)

## 4.7 CLLoginInterface Class Reference

Inheritance diagram for CLLoginInterface:



Collaboration diagram for CLLoginInterface:



### Public Member Functions

- void [showWelcomeMessage](#) ()  
*Show the login screen messages in the command line.*
- void [askEmployeeNumber](#) ()  
*Ask the user to input the employee number from the command line.*
- void [askEmployeeNumber](#) (const std::string &userNumber)  
*Default setter of employee number.*
- void [askNIF](#) ()  
*Ask the user to input the employee nif from the command line.*
- void [askNIF](#) (const std::string &nif)  
*Default setter of employee nif.*

## Additional Inherited Members

### 4.7.1 Member Function Documentation

#### 4.7.1.1 askEmployeeNumber() [1/2]

```
void CLLoginInterface::askEmployeeNumber ( ) [virtual]
```

Ask the user to input the employee number from the command line.

Implements [LoginInterface](#).

#### 4.7.1.2 askEmployeeNumber() [2/2]

```
void CLLoginInterface::askEmployeeNumber (
    const std::string & userNumber ) [virtual]
```

Default setter of employee number.

##### Note

This method is used when the user executes the main with the NIF and employee number

##### Parameters

<i>userNumber</i>	
-------------------	--

Implements [LoginInterface](#).

#### 4.7.1.3 askNIF() [1/2]

```
void CLLoginInterface::askNIF ( ) [virtual]
```

Ask the user to input the employee nif from the command line.

Implements [LoginInterface](#).

#### 4.7.1.4 askNIF() [2/2]

```
void CLLoginInterface::askNIF (
    const std::string & nif ) [virtual]
```

Default setter of employee nif.

##### Note

This method is used when the user executes the main with the NIF and employee number

##### Parameters

<i>nif</i>	
------------	--

Implements [LoginInterface](#).

#### 4.7.1.5 showWelcomeMessage()

```
void CLLoginInterface::showWelcomeMessage ( ) [virtual]
```

Show the login screen messages in the command line.

Implements [LoginInterface](#).

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

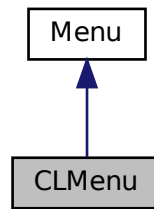
- include/CLLoginInterface.h
- src/CLLoginInterface.cpp

## 4.8 CLMenu Class Reference

Inheritance diagram for CLMenu:



Collaboration diagram for CLMenu:



## Public Member Functions

- void `show` (std::vector< `Sensor` \* >)  
*Shows the menu with six or less smaller sensors displayed in the command line.*
- void `show` (`Sensor` \*sensor)  
*Shows the menu for the sensor selected in a detailed format in the command line.*

## Protected Member Functions

- void `clearMenu` ()  
*Clear the menu in the command line.*

## Additional Inherited Members

### 4.8.1 Member Function Documentation

#### 4.8.1.1 `clearMenu()`

```
void CLMenu::clearMenu ( ) [protected]
```

Clear the menu in the command line.

#### 4.8.1.2 `show()` [1/2]

```
void CLMenu::show (
    Sensor * sensorToDisplay ) [virtual]
```

Shows the menu for the sensor selected in a detailed format in the command line.

## Parameters

<i>sensorToDisplay</i>	<a href="#">Sensor</a> to display
------------------------	-----------------------------------

Implements [Menu](#).

#### 4.8.1.3 show() [2/2]

```
void CLMenu::show (
    std::vector< Sensor * > sensors ) [virtual]
```

Shows the menu with six or less smaller sensors displayed in the command line.

## Parameters

<i>sensors</i>	Array of sensor to display <= 7
----------------	---------------------------------

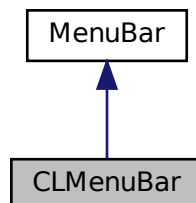
Implements [Menu](#).

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

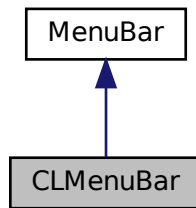
- include/[CLMenu.h](#)
- src/[CLMenu.cpp](#)

## 4.9 CLMenuBar Class Reference

Inheritance diagram for CLMenuBar:



Collaboration diagram for CLMenuBar:



## Public Member Functions

- void [show](#) ()

*Show the menu bar in the command line.*

## Additional Inherited Members

### 4.9.1 Member Function Documentation

#### 4.9.1.1 show()

```
void CLMenuBar::show ( ) [virtual]
```

Show the menu bar in the command line.

Implements [MenuBar](#).

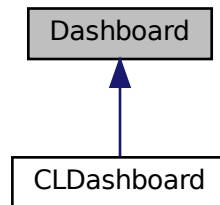
The documentation for this class was generated from the following files:

- include/[CLMenuBar.h](#)
- src/[CLMenuBar.cpp](#)

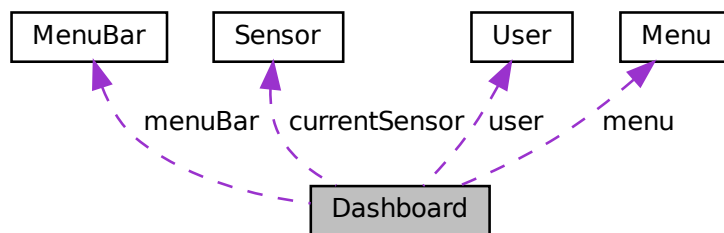


## 4.10 Dashboard Class Reference

Inheritance diagram for Dashboard:



Collaboration diagram for Dashboard:



### Public Member Functions

- virtual void **showMainMenu** ()=0
- void **changeInterface** (std::string newInterface="")  
*Changes the current interface to the input one.*
- void **addToMainMenu** ()  
*Adds all the current sensors to the dashboard.*
- void **addToMainMenu** (Sensor \*sensor)  
*Adds a sensor to the dashboard.*
- void **moveWindowMainMenu** (int n=1)  
*Shifts the main menu n pages.*
- void **changeCurrentSensorInfo** (std::string toChange, std::string newValue)  
*Changes the current sensor values.*
- void **setUser** (User user)  
*Adds the user to the dashboard.*
- void **addNewSensor** (std::string type)

- Creates the new sensor and calls addToMainMenu.*

  - void `deleteSensor` (std::string id)

*Deletes the desired sensor from the dashboard.*
- void `cleanSensor` ()

*Deletes all the sensors from the dashboard.*
- void `exit` ()

*Exits the dashboard and finishes the program.*
- void `logout` ()

*Exits the dashboard and goes back to the login screen.*
- bool `canExit` ()

*Called from main, checks if the program is allowed to exit.*

## Static Public Member Functions

- static `Dashboard * Create` (const std::string type="CLI")
- Create a new instance of the desired `Dashboard` interface.*

## Protected Attributes

- std::vector< `Sensor` \* > `sensor`
- Vector of all the vectors in the dashboard.*
- std::vector< std::vector< `Sensor` \* > > `mainMenu`
- Vetor that contains the pages of the main `Menu`.*
- int `mainMenuIndex`
- Current page of the main `Menu`.*
- `MenuBar` \* `menuBar`
  - `Menu` \* `menu`
  - `User` `user`
  - std::string `currentInterface`
- Current interface, can be a sensor ID or the main menu (..)*
- std::string `lastInterface`
- Last interface, can be a sensor ID or the main menu (..)*
- `Sensor` \* `currentSensor`
- If the currentInterface is a sensor then the value would be that sensor, if not the value is nullptr.*
- bool `allowedToExit`

## 4.10.1 Member Function Documentation

### 4.10.1.1 addNewSensor()

```
void Dashboard::addNewSensor (
    std::string type )
```

Creates the new sensor and calls addToMainMenu.

## Parameters

<i>type</i>	New sensor type
-------------	-----------------

## See also

[addToMainMenu](#)**4.10.1.2 addToMainMenu()** [1/2]

```
void Dashboard::addToMainMenu ( )
```

Adds all the current sensors to the dashboard.

## Note

Only called at the start of the program

**4.10.1.3 addToMainMenu()** [2/2]

```
void Dashboard::addToMainMenu (
    Sensor * sensor )
```

Adds a sensor to the dashboard.

## Parameters

<i>sensor</i>	<a href="#">Sensor</a> to add
---------------	-------------------------------

**4.10.1.4 canExit()**

```
bool Dashboard::canExit ( )
```

Called from main, checks if the program is allowed to exit.

## Returns

true End the program  
false Cannot exit, goes to login screen

#### 4.10.1.5 `changeCurrentSensorInfo()`

```
void Dashboard::changeCurrentSensorInfo (
    std::string toChange,
    std::string newValue )
```

Changes the current sensor values.

##### Parameters

<i>toChange</i>	Desired value to change
<i>newValue</i>	New value to add

#### 4.10.1.6 `changeInterface()`

```
void Dashboard::changeInterface (
    std::string newInterface = "" )
```

Changes the current interface to the input one.

##### Parameters

<i>newInterface</i>	Desired interface to change ( .. = main <a href="#">Menu</a> )
---------------------	--

#### 4.10.1.7 `cleanSensor()`

```
void Dashboard::cleanSensor ( )
```

Deletes all the sensors from the dashboard.

##### Note

Only called when exiting the dashboard

#### 4.10.1.8 `Create()`

```
Dashboard * Dashboard::Create (
    const std::string type = "CLI" ) [static]
```

Create a new instance of the desired [Dashboard](#) interface.

## Parameters

<i>type</i>	Type of <a href="#">Dashboard</a> interface
-------------	---

## Returns

Dashboard\*

**4.10.1.9 deleteSensor()**

```
void Dashboard::deleteSensor (
    std::string id )
```

Deletes the desired sensor from the dashboard.

## Parameters

<i>id</i>	ID of the sensor to delete
-----------	----------------------------

## See also

[addToMainMenu](#)

**4.10.1.10 exit()**

```
void Dashboard::exit ( )
```

Exits the dashboard and finishes the program.

## See also

[cleanSensor](#)

**4.10.1.11 logout()**

```
void Dashboard::logout ( )
```

Exits the dashboard and goes back to the login screen.

## See also

[cleanSensor](#)

#### 4.10.1.12 moveWindowMainMenu()

```
void Dashboard::moveWindowMainMenu (
    int n = 1 )
```

Shifts the main menu n pages.

##### Parameters

<i>n</i>	Number of pages to shift ( positive to the right )
----------	--

#### 4.10.1.13 setUser()

```
void Dashboard::setUser (
    User user )
```

Adds the user to the dashboard.

##### Parameters

<i>user</i>	Current user
-------------	--------------

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

- include/[Dashboard.h](#)
- src/[Dashboard.cpp](#)

## 4.11 Database Class Reference

### Public Member Functions

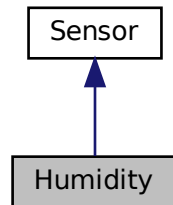
- void **addUser** ([User](#))
- [User](#) **getUser** (std::string, std::string)

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

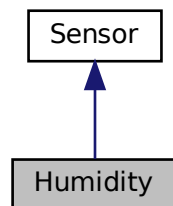
- include/[Database.h](#)
- src/[Database.cpp](#)

## 4.12 Humidity Class Reference

Inheritance diagram for Humidity:



Collaboration diagram for Humidity:



### Public Member Functions

- `Humidity` (bool active=false, int valPerMin=1)  
*Create a new `Humidity` object.*
- `std::vector< int > requestData ()`  
*Request data from the humidity sensor.*

### Additional Inherited Members

#### 4.12.1 Constructor & Destructor Documentation

##### 4.12.1.1 Humidity()

```
Humidity::Humidity (  
    bool active = false,  
    int valPerMin = 1 )
```

Create a new `Humidity` object.

**Parameters**

<i>active</i>	By default off
<i>valPerMin</i>	By default 1 val/min

## 4.12.2 Member Function Documentation

### 4.12.2.1 requestData()

```
std::vector< int > Humidity::requestData ( ) [virtual]
```

Request data from the humidity sensor.

**Returns**

std::vector<int> Data values

**Note**

This function is a placeholder, returns random values

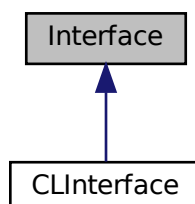
Implements [Sensor](#).

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

- include/[Humidity.h](#)
- src/[Humidity.cpp](#)

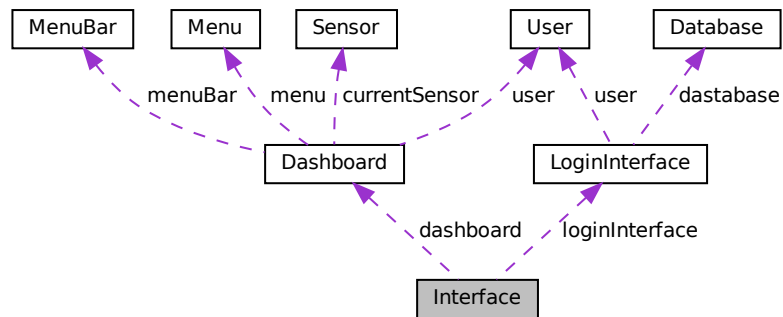
## 4.13 Interface Class Reference

Inheritance diagram for Interface:





Collaboration diagram for Interface:



## Public Member Functions

- virtual void **login** (int tries)=0
- void **login** (std::string userNumber, std::string nif)  
*Default login method withouth any interface.*
- bool **loadMenu** ()  
*Creates and loads the dashboard.*

## Static Public Member Functions

- static **Interface** \* **Create** (const std::string type="CLI")  
*Create a new **Interface** instance of the type specified.*

## Protected Attributes

- **LoginInterface** \* **loginInterface**
- **Dashboard** \* **dashboard**

### 4.13.1 Member Function Documentation

#### 4.13.1.1 Create()

```

Interface * Interface::Create (
    const std::string type = "CLI" ) [static]
  
```

Create a new **Interface** instance of the type specified.

**Parameters**

<i>type</i>	Type of interface
-------------	-------------------

**Returns**

Interface\* New [Interface](#) instance

**4.13.1.2 loadMenu()**

```
bool Interface::loadMenu ( )
```

Creates and loads the dashboard.

**Returns**

true = exit

false = login again

**4.13.1.3 login()**

```
void Interface::login (
    std::string userNumber,
    std::string nif )
```

Default login method without any interface.

**Parameters**

<i>userNumber</i>	
<i>nif</i>	

**Note**

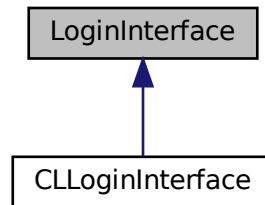
This method is used when the user executes the main with the NIF and employee number

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

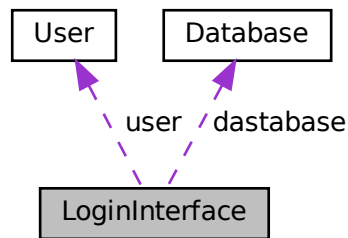
- [include/Interface.h](#)
- [src/Interface.cpp](#)

## 4.14 LoginInterface Class Reference

Inheritance diagram for LoginInterface:



Collaboration diagram for LoginInterface:



### Public Member Functions

- virtual void **showWelcomeMessage** ()=0
- virtual void **askEmployeeNumber** ()=0
- virtual void **askEmployeeNumber** (const std::string &userNumber)=0
- virtual void **askNIF** ()=0
- virtual void **askNIF** (const std::string &userNumber)=0
- bool **checkUser** ()

*Check if the user data corresponds to a real user and stores the user.*

- **User** **getUser** ()

*Returns the user that has previously logged in.*

### Static Public Member Functions

- static **LoginInterface** \* **Create** (const std::string="CLI")

*Create a new Login interface instance of the type specified.*

## Protected Attributes

- `std::string inputNIF`
- `std::string inputEmployeeNumber`
- `User user`
- `Database dastabase`

### 4.14.1 Member Function Documentation

#### 4.14.1.1 `checkUser()`

```
bool LoginInterface::checkUser ( )
```

Check if the user data corresponds to a real user and stores the user.

##### Returns

`true` = The user exists

`false` = The user does not exists

#### 4.14.1.2 `Create()`

```
LoginInterface * LoginInterface::Create (
    const std::string type = "CLI" ) [static]
```

Create a new Login interface instance of the type specified.

##### Parameters

<i>type</i>	Type of login interface
-------------	-------------------------

##### Returns

`LoginInterface*` New login interface instance

#### 4.14.1.3 `getUser()`

```
User LoginInterface::getUser ( )
```

Returns the user that has previously logged in.

## Returns

[User](#)

## See also

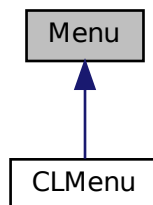
[checkUser\(\)](#)

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

- include/[LoginInterface.h](#)
- src/[LoginInterface.cpp](#)

## 4.15 Menu Class Reference

Inheritance diagram for Menu:



### Public Member Functions

- virtual void **show** (std::vector< [Sensor](#) \* >)=0
- virtual void **show** ([Sensor](#) \*)=0

### Static Public Member Functions

- static [Menu](#) \* **Create** (const std::string type="CLI")  
*Create a new [Menu](#) instance of the type specified.*

#### 4.15.1 Member Function Documentation

##### 4.15.1.1 Create()

```
Menu * Menu::Create (  
    const std::string type = "CLI" ) [static]
```

Create a new [Menu](#) instance of the type specified.

#### Parameters

<i>type</i>	Type of menu
-------------	--------------

#### Returns

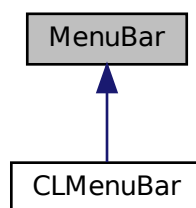
Menu\* New menu instance

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

- [include/Menu.h](#)
- [src/Menu.cpp](#)

## 4.16 MenuBar Class Reference

Inheritance diagram for MenuBar:



### Public Member Functions

- virtual void **show** ()=0
- void [setUserName](#) (std::string userName)  
*Sets the user name.*
- void [setCurrentMenu](#) (std::string currentMenu)  
*Sets the current menu.*

### Static Public Member Functions

- static [MenuBar](#) \* [Create](#) (const std::string type="CLI")  
*Create a new [Menu](#) bar instance of the type specified.*

### Protected Attributes

- std::string **userName**
- std::string **currentMenu**

## 4.16.1 Member Function Documentation

### 4.16.1.1 Create()

```
MenuBar * MenuBar::Create (
    const std::string type = "CLI" ) [static]
```

Create a new [Menu](#) bar instance of the type specified.

#### Parameters

<i>type</i>	Type of menu bar
-------------	------------------

#### Returns

MenuBar\* New menu bar instance

### 4.16.1.2 setCurrentMenu()

```
void MenuBar::setCurrentMenu (
    std::string currentMenu )
```

Sets the current menu.

#### Parameters

<i>currentMenu</i>	
--------------------	--

### 4.16.1.3 setUsername()

```
void MenuBar::setUserName (
    std::string userName )
```

Sets the user name.

#### Parameters

<i>userName</i>	
-----------------	--

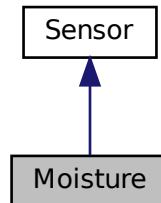
The documentation for this class was generated from the following files:

- include/[MenuBar.h](#)

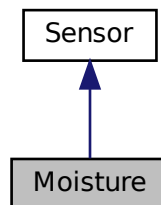
- [src/MenuBar.cpp](#)

## 4.17 Moisture Class Reference

Inheritance diagram for Moisture:



Collaboration diagram for Moisture:



### Public Member Functions

- [Moisture](#) (bool active=false, int valPerMin=1)  
*Create a new [Moisture](#) object.*
- `std::vector< int > requestData ()`  
*Request data from the moisture sensor.*

### Additional Inherited Members

#### 4.17.1 Constructor & Destructor Documentation



### 4.17.1.1 Moisture()

```
Moisture::Moisture (
    bool active = false,
    int valPerMin = 1 )
```

Create a new [Moisture](#) object.

#### Parameters

<i>active</i>	By default off
<i>valPerMin</i>	By default 1 val/min

## 4.17.2 Member Function Documentation

### 4.17.2.1 requestData()

```
std::vector< int > Moisture::requestData ( ) [virtual]
```

Request data from the moisture sensor.

#### Returns

std::vector<int> Data values

#### Note

This function is a placeholder, returns random values

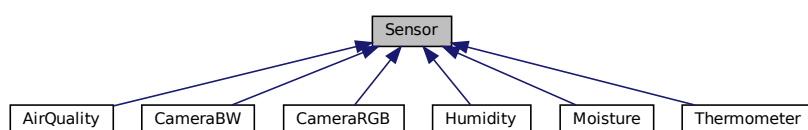
Implements [Sensor](#).

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

- include/[Moisture.h](#)
- src/[Moisture.cpp](#)

## 4.18 Sensor Class Reference

Inheritance diagram for Sensor:



## Public Member Functions

- [Sensor](#) (std::string id="0000", std::string type="none", std::string magnitude="-", bool active=true, int valPerMin=1)  
Construct a new [Sensor::Sensor](#) object.
- std::string [getId](#) ()  
Returns the sensor id.
- std::string [getType](#) ()  
Returns the sensor type.
- void [setActive](#) (bool)  
Turns the sensor On/Off.
- bool [isActive](#) ()  
Returns if the sensor is On/Off.
- void [setArea](#) (std::string)  
Set the area of the sensor.
- std::string [getArea](#) ()  
Returns the area of the sensor.
- void [setMagnitude](#) (std::string magnitude)  
Set the sensor data values magnitude.
- std::string [getMagnitude](#) ()  
Returns the sensor magnitude.
- void [setValPerMin](#) (int valPerMin)  
Set the number of values per min.
- int [getValPerMin](#) ()  
Return the number of values per min.
- virtual std::vector< int > [requestData](#) ()=0

## Static Public Member Functions

- static [Sensor](#) \* [Create](#) (const std::string type)  
Creates an instance of the sensor type.

## Protected Member Functions

- void [addNewData](#) (int)
- void [removeOldData](#) (int)

## Protected Attributes

- std::string [id](#)
- bool [active](#)
- std::string [type](#)
- std::string [area](#)
- std::vector< int > [data](#)
- std::string [magnitude](#)
- int [valPerMin](#)

### 4.18.1 Constructor & Destructor Documentation

#### 4.18.1.1 Sensor()

```
Sensor::Sensor (
    std::string id = "0000",
    std::string type = "none",
    std::string magnitude = "-",
    bool active = true,
    int valPerMin = 1 )
```

Construct a new [Sensor](#):: [Sensor](#) object.

##### Parameters

<i>id</i>	Identification of type and number
<i>type</i>	Type of sensor
<i>magnitude</i>	Data values magnitude
<i>active</i>	<a href="#">Sensor</a> is On/Off
<i>valPerMin</i>	Number of values per min

### 4.18.2 Member Function Documentation

#### 4.18.2.1 Create()

```
Sensor * Sensor::Create (
    const std::string type ) [static]
```

Creates an instance of the sensor type.

##### Parameters

<i>type</i>	
-------------	--

##### Returns

[Sensor](#)\*

#### 4.18.2.2 getArea()

```
std::string Sensor::getArea ( )
```

Returns the area of the sensor.

##### Returns

std::string Area

#### 4.18.2.3 getId()

```
std::string Sensor::getId ( )
```

Returns the sensor id.

##### Returns

std::string ID

#### 4.18.2.4 getMagnitude()

```
std::string Sensor::getMagnitude ( )
```

Returns the sensor magnitude.

##### Returns

std::string Magnitude

#### 4.18.2.5 getType()

```
std::string Sensor::getType ( )
```

Returns the sensor type.

##### Returns

std::string [Sensor](#) type

#### 4.18.2.6 getValPerMin()

```
int Sensor::getValPerMin ( )
```

Return the number of values per min.

##### Returns

int Values per minute

#### 4.18.2.7 isActive()

```
bool Sensor::isActive ( )
```

Returns if the sensor is On/Off.

##### Returns

true = ON

false = OFF

#### 4.18.2.8 setActive()

```
void Sensor::setActive (
    bool active )
```

Turns the sensor On/Off.

##### Parameters

<i>active</i>	
---------------	--

#### 4.18.2.9 setArea()

```
void Sensor::setArea (
    std::string area )
```

Set the area of the sensor.

##### Parameters

<i>area</i>	
-------------	--

#### 4.18.2.10 setMagnitude()

```
void Sensor::setMagnitude (
    std::string magnitude )
```

Set the sensor data values magnitude.

##### Parameters

<i>magnitude</i>	
------------------	--

#### 4.18.2.11 setValPerMin()

```
void Sensor::setValPerMin (
    int valPerMin )
```

Set the number of values per min.

##### Parameters

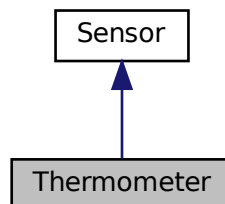
<i>valPerMin</i>	
------------------	--

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

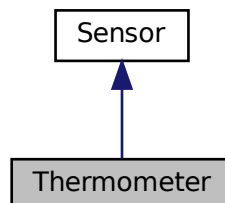
- include/[Sensor.h](#)
- src/[Sensor.cpp](#)

## 4.19 Thermometer Class Reference

Inheritance diagram for Thermometer:



Collaboration diagram for Thermometer:



## Public Member Functions

- [Thermometer](#) (bool active=false, int valPerMin=1)  
*Create a new [Thermometer](#) object.*
- std::vector< int > [requestData](#) ()  
*Request data from the thermometer.*

## Additional Inherited Members

### 4.19.1 Constructor & Destructor Documentation

#### 4.19.1.1 Thermometer()

```
Thermometer::Thermometer (
    bool active = false,
    int valPerMin = 1 )
```

Create a new [Thermometer](#) object.

##### Parameters

<i>active</i>	By default off
<i>valPerMin</i>	By default 1 val/min

### 4.19.2 Member Function Documentation

#### 4.19.2.1 requestData()

```
std::vector< int > Thermometer::requestData ( ) [virtual]
```

Request data from the thermometer.

##### Returns

std::vector<int> Data values

##### Note

This function is a placeholder, returns random values

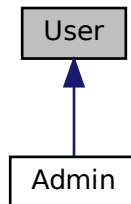
Implements [Sensor](#).

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

- include/[Thermometer.h](#)
- src/[Thermometer.cpp](#)

## 4.20 User Class Reference

Inheritance diagram for User:



### Public Member Functions

- [User](#) (std::string number="00000", std::string nif="00000000", std::string name="default")  
*Create a new [User](#).*
- bool [isSameNIF](#) (std::string)  
*Check if the user has the input NIF.*
- bool [isSameEmployeeNumber](#) (std::string)  
*Check if the user has the input employee number.*
- std::string [getName](#) ()  
*Returns the user name.*
- void [addTimestamp](#) ()  
*Set the timestamp.*
- std::string [getTimestamp](#) ()  
*Returns the timestamp.*
- bool [hasAdminPermission](#) ()  
*Returns the user permissions.*

### Protected Member Functions

- void [setNIF](#) (std::string)  
*Set the user NIF.*
- void [setEmployeeNumber](#) (std::string)  
*Set the user employee number.*

### Protected Attributes

- std::string **name**
- std::string **NIF**
- std::string **employeeNumber**
- std::string **lastLogTime**
- bool **adminPermission**



## 4.20.1 Constructor & Destructor Documentation

### 4.20.1.1 User()

```
User::User (
    std::string employeeNumber = "00000",
    std::string NIF = "00000000",
    std::string name = "default" )
```

Create a new [User](#).

#### Parameters

<i>employeeNumber</i>	
<i>NIF</i>	
<i>name</i>	

## 4.20.2 Member Function Documentation

### 4.20.2.1 addTimestamp()

```
void User::addTimestamp ( )
```

Set the timestamp.

### 4.20.2.2 getName()

```
std::string User::getName ( )
```

Returns the user name.

#### Returns

std::string [User](#) name

#### 4.20.2.3 `getTimestamp()`

```
std::string User::getTimestamp ( )
```

Returns the timestamp.

##### Returns

std::string Timestamp

#### 4.20.2.4 `hasAdminPermission()`

```
bool User::hasAdminPermission ( )
```

Returns the user permissions.

##### Returns

true = has admin permissions

false = does not have admin permissions

#### 4.20.2.5 `isSameEmployeeNumber()`

```
bool User::isSameEmployeeNumber (
    std::string employeeNumber )
```

Check if the user has the input employee number.

##### Parameters

<i>employeeNumber</i>	
-----------------------	--

##### Returns

true = is the same employee number

false = is not the same employee number

#### 4.20.2.6 `isSameNIF()`

```
bool User::isSameNIF (
    std::string NIF )
```

Check if the user has the input NIF.

## Parameters

$N \leftrightarrow$	
<i>IF</i>	

## Returns

true = is the same NIF

false = is not the same NIF

**4.20.2.7 setEmployeeNumber()**

```
void User::setEmployeeNumber (
    std::string employeeNumber ) [protected]
```

Set the user employee number.

## Parameters

<i>employeeNumber</i>	
-----------------------	--

**4.20.2.8 setNIF()**

```
void User::setNIF (
    std::string NIF ) [protected]
```

Set the user NIF.

## Parameters

$N \leftrightarrow$	
<i>IF</i>	

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

- [include/User.h](#)
- [src/User.cpp](#)



## Chapter 5

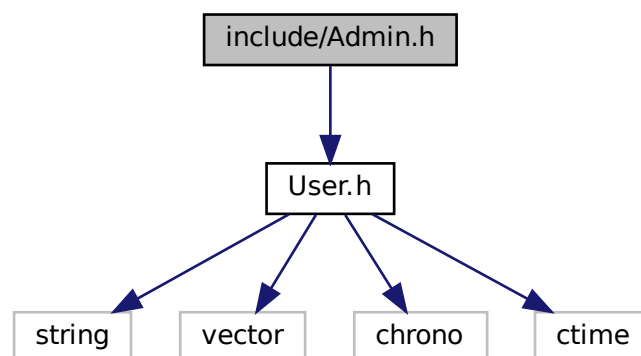
# File Documentation

### 5.1 include/Admin.h File Reference

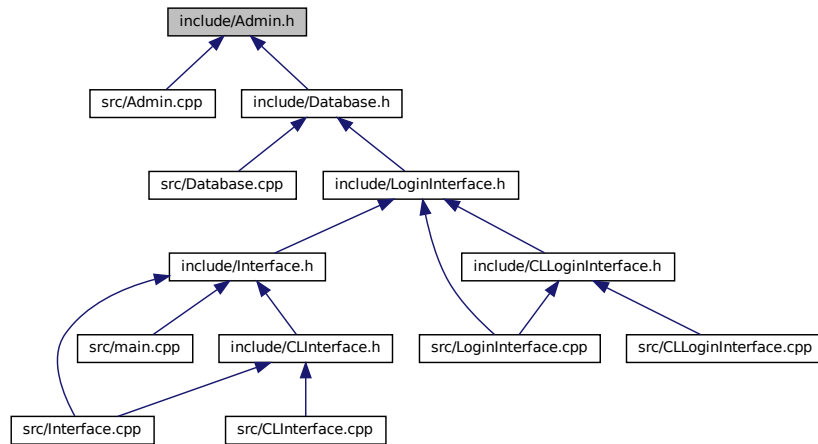
Header file of [Admin.cpp](#).

```
#include "User.h"
```

Include dependency graph for Admin.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [Admin](#)

### 5.1.1 Detailed Description

Header file of [Admin.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

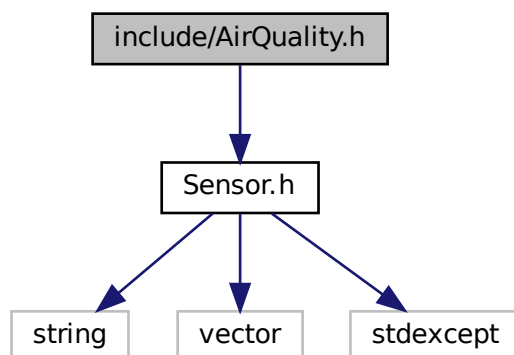
Copyright (c) 2022

## 5.2 include/AirQuality.h File Reference

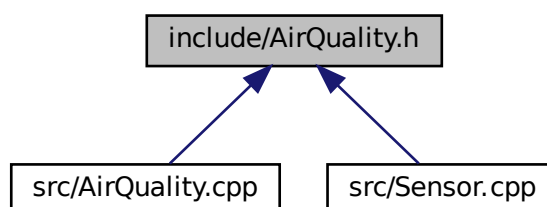
Header file of [AirQuality.cpp](#).

```
#include "Sensor.h"
```

Include dependency graph for AirQuality.h:



This graph shows which files directly or indirectly include this file:



### Classes

- class [AirQuality](#)

#### 5.2.1 Detailed Description

Header file of [AirQuality.cpp](#).

**Author**

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

**Version**

1.0

**Date**

2022-11-23

**Copyright**

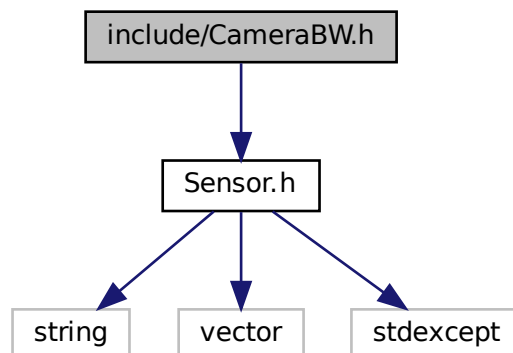
Copyright (c) 2022

### 5.3 include/CameraBW.h File Reference

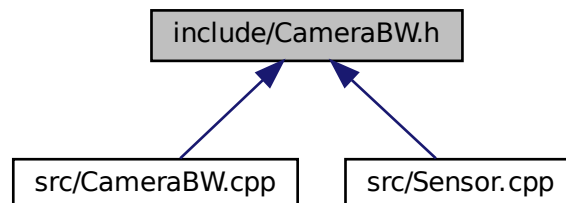
Header file of [CameraBW.cpp](#).

```
#include "Sensor.h"
```

Include dependency graph for CameraBW.h:



This graph shows which files directly or indirectly include this file:





## Classes

- class [CameraBW](#)

### 5.3.1 Detailed Description

Header file of [CameraBW.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

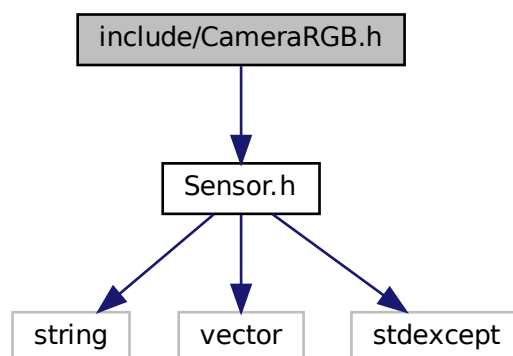
Copyright (c) 2022

## 5.4 include/CameraRGB.h File Reference

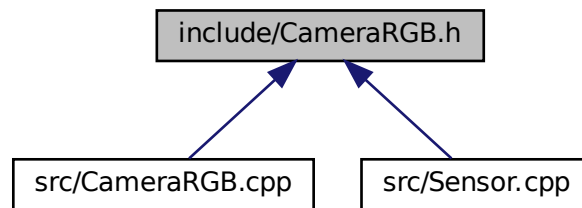
Header file of [CameraRGB.cpp](#).

```
#include "Sensor.h"
```

Include dependency graph for CameraRGB.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [CameraRGB](#)

### 5.4.1 Detailed Description

Header file of [CameraRGB.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

Copyright (c) 2022

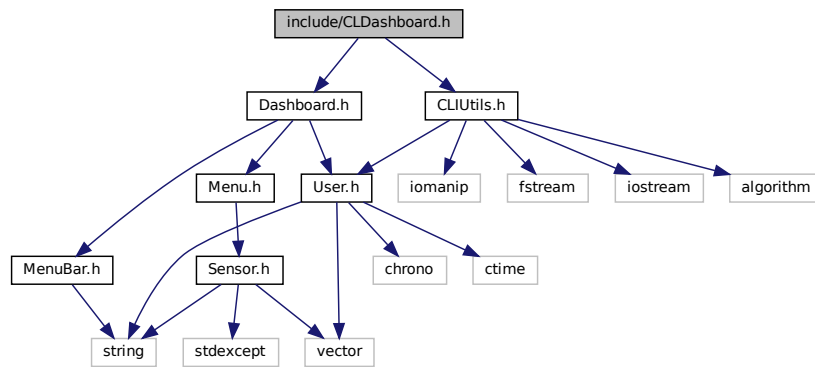
## 5.5 include/CLDashboard.h File Reference

Header file of [CLDashboard.cpp](#).

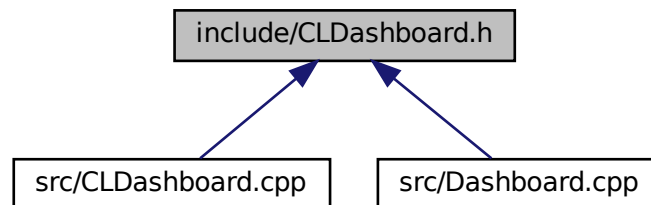
```
#include "Dashboard.h"
```

```
#include "CLIUtils.h"
```

Include dependency graph for CLDashboard.h:



This graph shows which files directly or indirectly include this file:



### Classes

- class [CLDashboard](#)

### 5.5.1 Detailed Description

Header file of [CLDashboard.cpp](#).

Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

## Version

1.0

## Date

2022-11-23

## Copyright

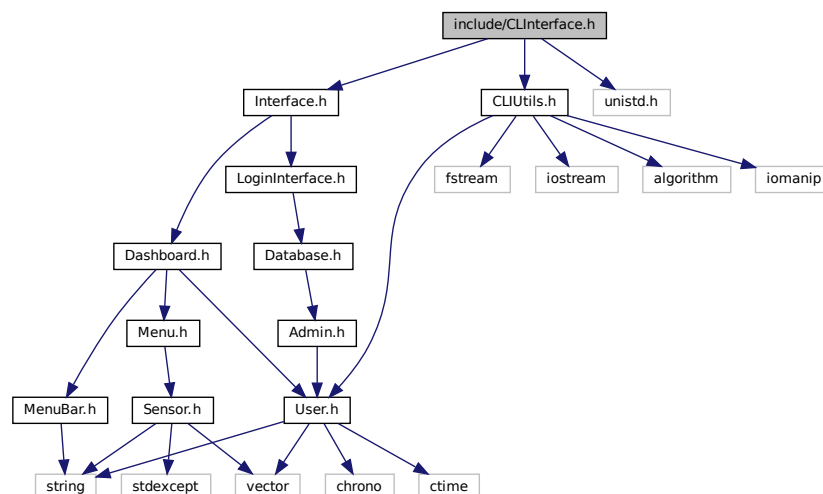
Copyright (c) 2022

## 5.6 include/CLInterface.h File Reference

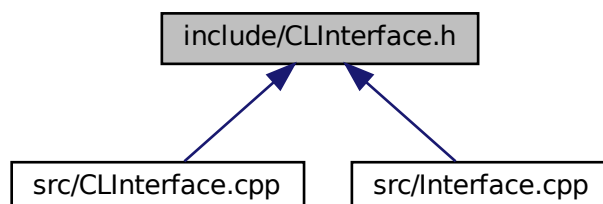
Header file of [CLInterface.cpp](#).

```
#include "Interface.h"
#include "CLIUtils.h"
#include <unistd.h>
```

Include dependency graph for CLInterface.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [CLIInterface](#)

### 5.6.1 Detailed Description

Header file of [CLIInterface.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

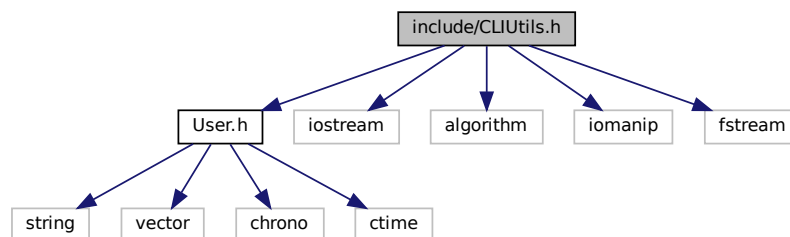
Copyright (c) 2022

## 5.7 include/CLIUtils.h File Reference

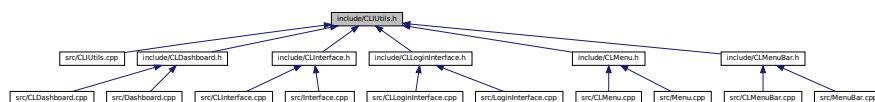
Header file of [CLIUtils.cpp](#).

```
#include "User.h"
#include <iostream>
#include <algorithm>
#include <iomanip>
#include <fstream>
```

Include dependency graph for CLIUtils.h:



This graph shows which files directly or indirectly include this file:



## Functions

- void **setTerminalSize** ()
- int **getTerminalWidth** ()
- int **getTerminalHeight** ()
- void **printCenterFromFile** (const std::string fileName, const std::string color="none")  
*Print the contents of the file in the center.*
- void **printCenter** (const std::string toPrint, const int padding=0)  
*Print the string in the center.*
- void **printCenter** (const std::string toPrint, const std::string color, const int padding=0)  
*Print the string in the center.*
- void **printRight** (const std::string toPrint, const int padding=0)
- void **printRight** (const std::string toPrint, const std::string color, const int padding=0)
- void **printLeft** (const std::string toPrint, const int padding=0)
- void **printLeft** (const std::string toPrint, const std::string color, const int padding=0)
- void **printColor** (std::string, std::string="none")  
*Print the string to the terminal in the given color.*
- std::string **setColor** (std::string)  
*Set the Color to print.*
- void **startCustomTerminal** (int)
- std::vector< std::string > **newCommand** (User &user, std::string="")
- void **clearCustomTerminal** (int)
- void **moveCursor** (int posX, int posY)
- void **printGraphic** (const std::vector< int > &data, int valPerY=1, int posX=10, int posY=10, int scale=1)

### 5.7.1 Detailed Description

Header file of [CLIUtils.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

Copyright (c) 2022

### 5.7.2 Function Documentation

#### 5.7.2.1 **printCenter()** [1/2]

```
void printCenter (
    const std::string toPrint,
    const int padding )
```

Print the string in the center.

## Parameters

<i>toPrint</i>	
<i>padding</i>	

**5.7.2.2 printCenter()** [2/2]

```
void printCenter (
    const std::string toPrint,
    const std::string color,
    const int padding )
```

Print the string in the center.

## Parameters

<i>toPrint</i>	
<i>padding</i>	
<i>color</i>	

**5.7.2.3 printCenterFromFile()**

```
void printCenterFromFile (
    std::string fileName,
    std::string color )
```

Print the contents of the file in the center.

## Parameters

<i>fileName</i>	
<i>color</i>	

**5.7.2.4 printColor()**

```
void printColor (
    std::string toPrint,
    std::string color )
```

Print the string to the terminal in the given color.

## Parameters

<i>toPrint</i>	
<i>color</i>	

## 5.7.2.5 setColor()

```
std::string setColor (
    std::string color )
```

Set the Color to print.

## Parameters

<i>color</i>	
--------------	--

## Returns

std::string

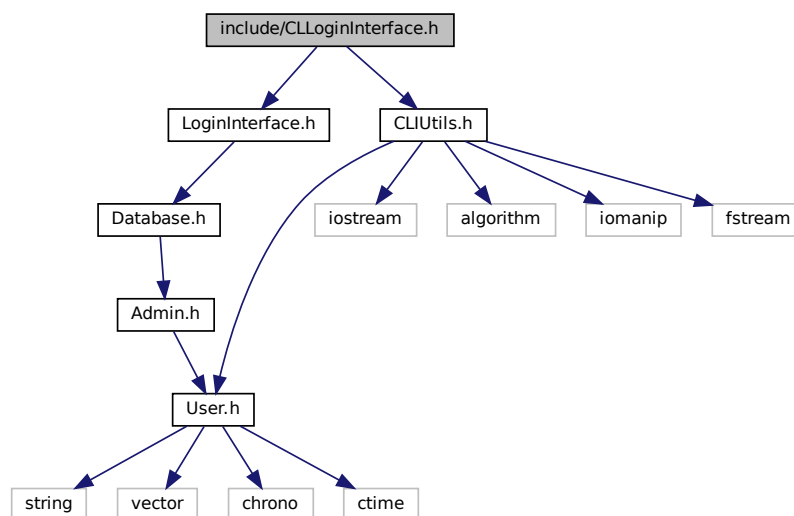
## 5.8 include/CLoginInterface.h File Reference

Header file of [CLoginInterface.cpp](#).

```
#include "LoginInterface.h"
```

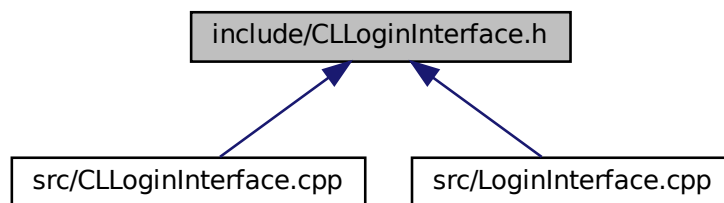
```
#include "CLIUtils.h"
```

Include dependency graph for CLoginInterface.h:





This graph shows which files directly or indirectly include this file:



## Classes

- class [CLLoginInterface](#)

### 5.8.1 Detailed Description

Header file of [CLLoginInterface.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

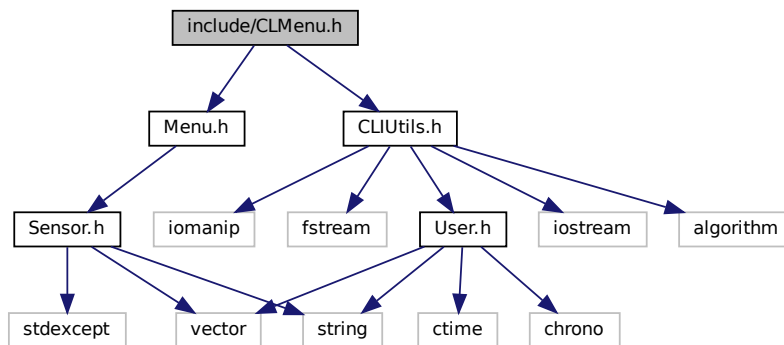
#### Copyright

Copyright (c) 2022

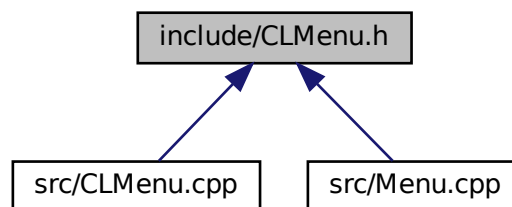
## 5.9 include/CLMenu.h File Reference

Header file of [CLMenu.cpp](#).

```
#include "Menu.h"
#include "CLIUtils.h"
Include dependency graph for CLMenu.h:
```



This graph shows which files directly or indirectly include this file:



### Classes

- class [CLMenu](#)

### 5.9.1 Detailed Description

Header file of [CLMenu.cpp](#).

Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

## Version

1.0

## Date

2022-11-23

## Copyright

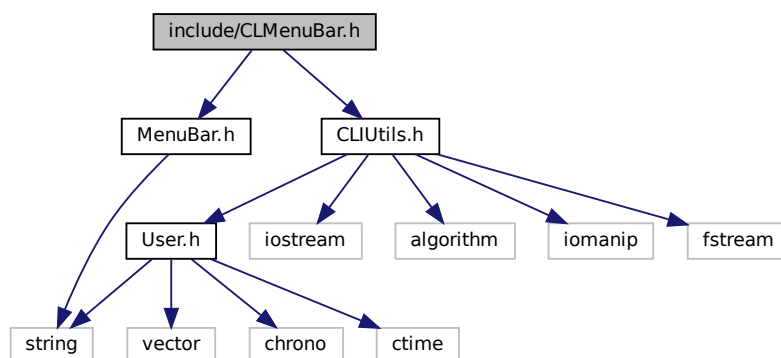
Copyright (c) 2022

## 5.10 include/CLMenuBar.h File Reference

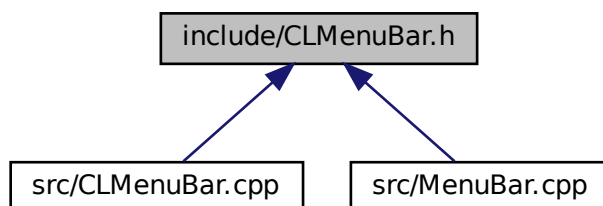
Header file of [CLMenuBar.cpp](#).

```
#include "MenuBar.h"  
#include "CLIUtils.h"
```

Include dependency graph for CLMenuBar.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [CLMenuBar](#)

### 5.10.1 Detailed Description

Header file of [CLMenuBar.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

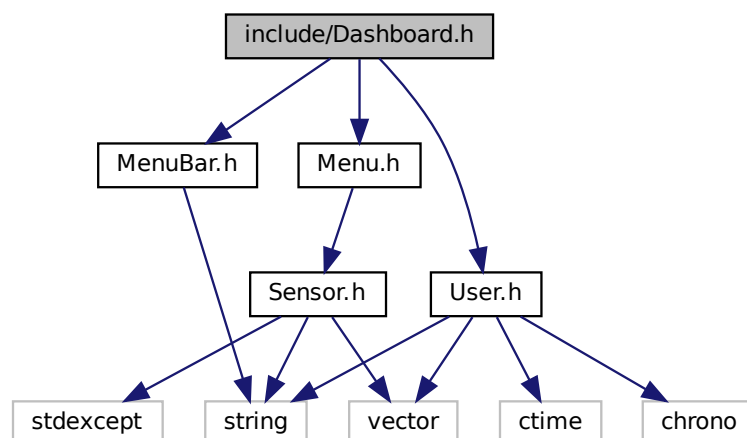
Copyright (c) 2022

## 5.11 include/Dashboard.h File Reference

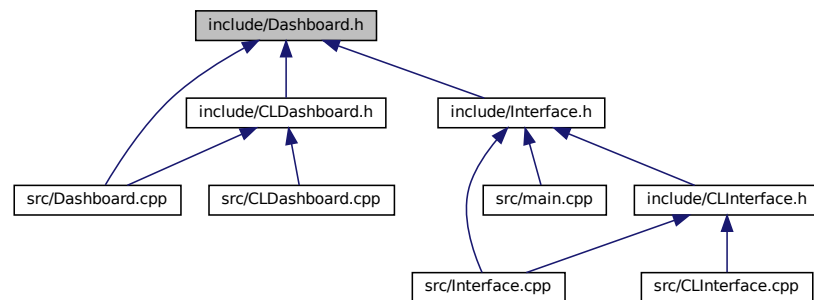
Header file of [Dashboard.cpp](#).

```
#include "User.h"  
#include "MenuBar.h"  
#include "Menu.h"
```

Include dependency graph for Dashboard.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [Dashboard](#)

### 5.11.1 Detailed Description

Header file of [Dashboard.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

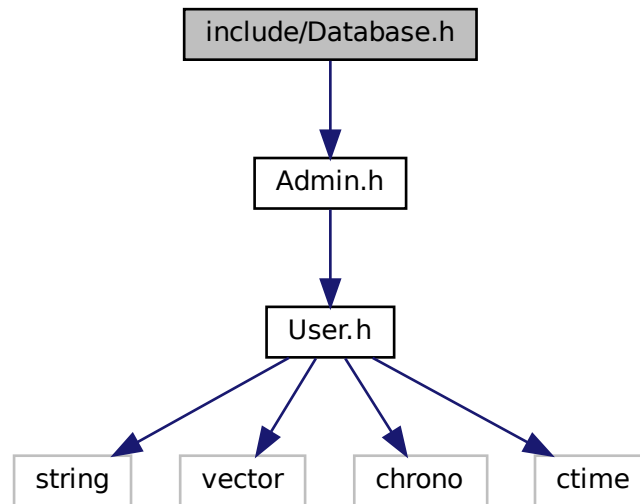
Copyright (c) 2022

## 5.12 include/Database.h File Reference

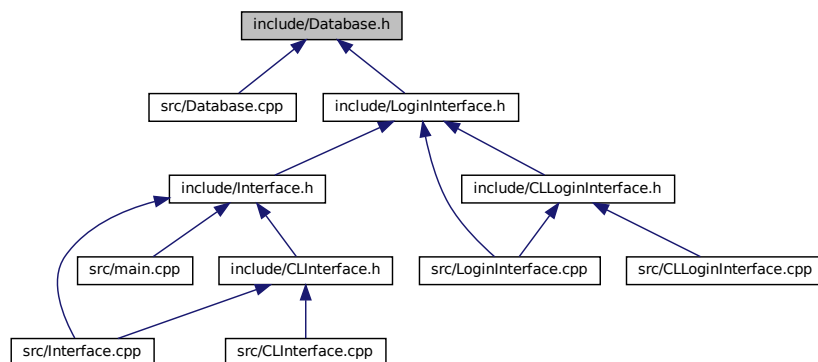
Header file of [Database.cpp](#).

```
#include "Admin.h"
```

Include dependency graph for Database.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [Database](#)

### 5.12.1 Detailed Description

Header file of [Database.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

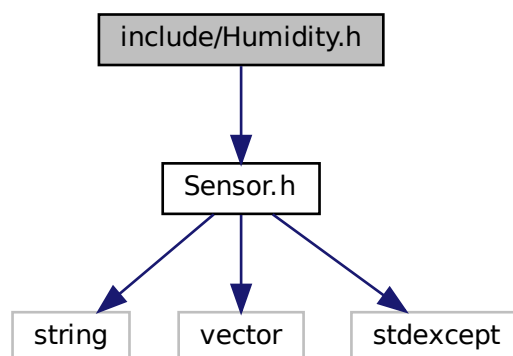
Copyright (c) 2022

## 5.13 include/Humidity.h File Reference

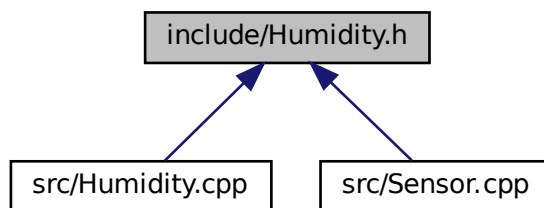
Header file of [Humidity.cpp](#).

```
#include "Sensor.h"
```

Include dependency graph for Humidity.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [Humidity](#)

### 5.13.1 Detailed Description

Header file of [Humidity.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

Copyright (c) 2022

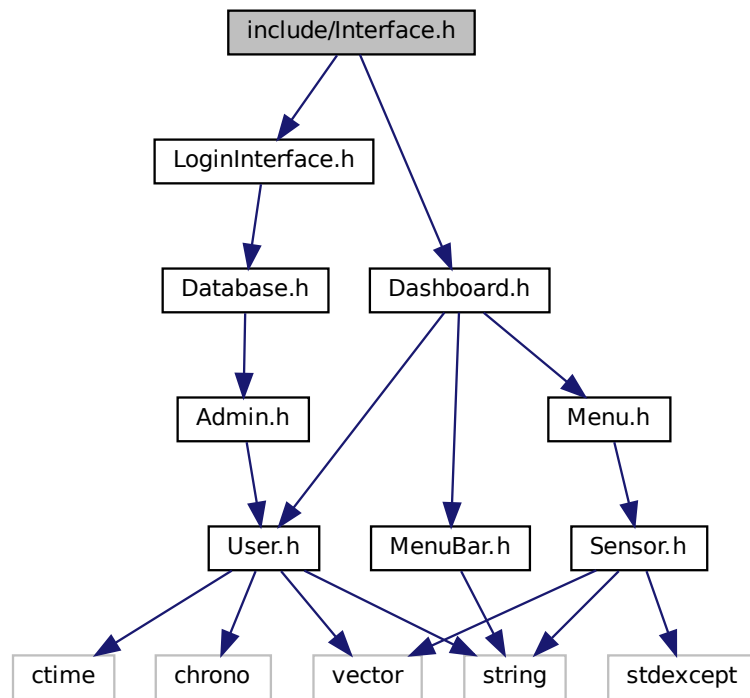


## 5.14 include/Interface.h File Reference

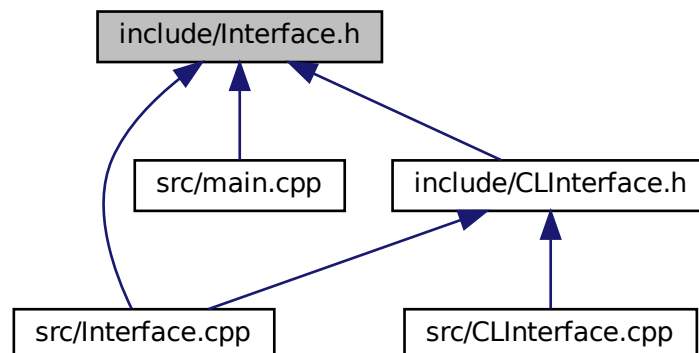
Header file of [Interface.cpp](#).

```
#include "LoginInterface.h"  
#include "Dashboard.h"
```

Include dependency graph for Interface.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [Interface](#)

### 5.14.1 Detailed Description

Header file of [Interface.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

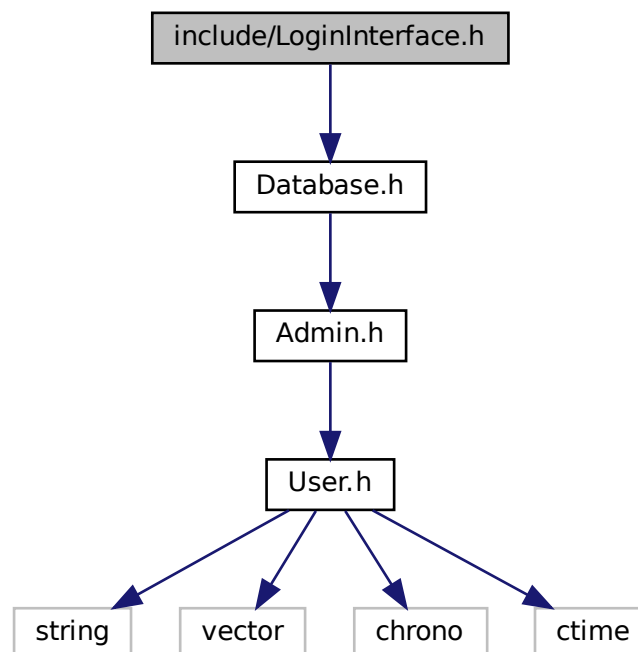
Copyright (c) 2022

## 5.15 include/LoginInterface.h File Reference

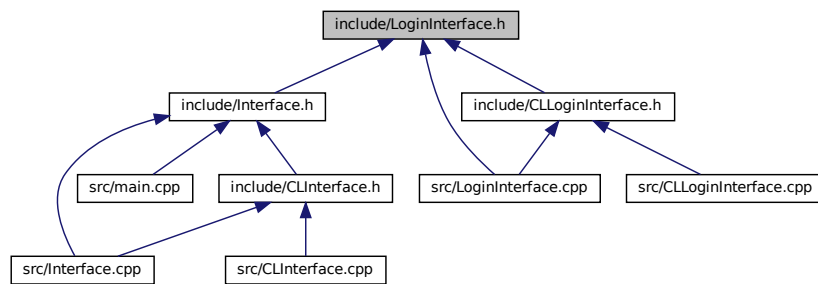
Header file of [LoginInterface.cpp](#).

```
#include "Database.h"
```

Include dependency graph for LoginInterface.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [LoginInterface](#)

### 5.15.1 Detailed Description

Header file of [LoginInterface.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

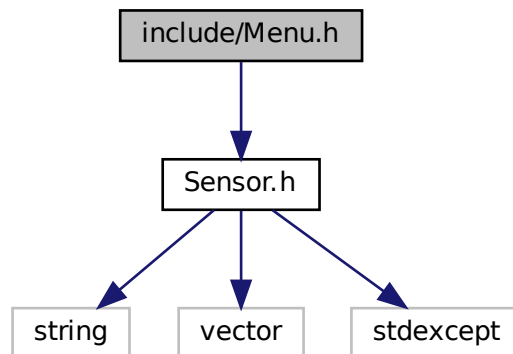
Copyright (c) 2022

## 5.16 include/Menu.h File Reference

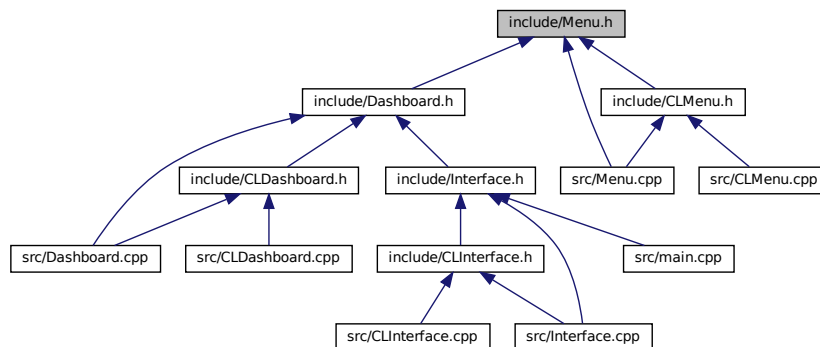
Header file of [Menu.cpp](#).

```
#include "Sensor.h"
```

Include dependency graph for Menu.h:



This graph shows which files directly or indirectly include this file:



### Classes

- class [Menu](#)

#### 5.16.1 Detailed Description

Header file of [Menu.cpp](#).

## Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

## Version

1.0

## Date

2022-11-23

## Copyright

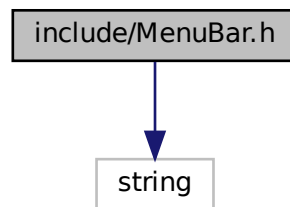
Copyright (c) 2022

## 5.17 include/MenuBar.h File Reference

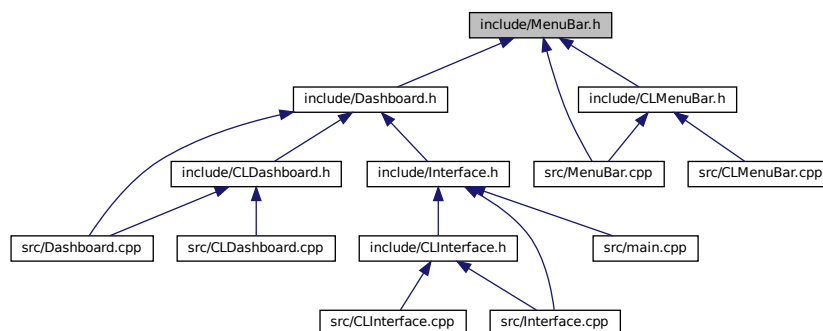
Header file of [MenuBar.cpp](#).

```
#include <string>
```

Include dependency graph for MenuBar.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [MenuBar](#)

### 5.17.1 Detailed Description

Header file of [MenuBar.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

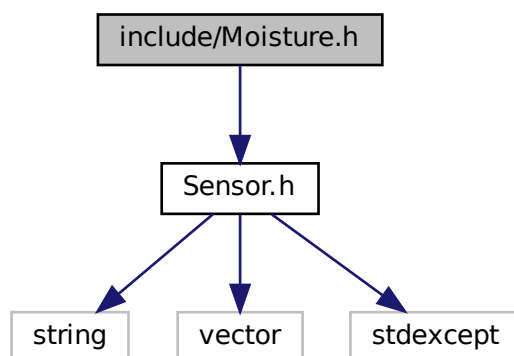
Copyright (c) 2022

## 5.18 include/Moisture.h File Reference

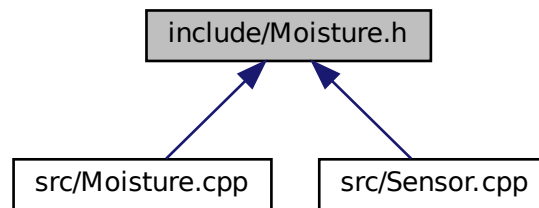
Header file of [Moisture.cpp](#).

```
#include "Sensor.h"
```

Include dependency graph for Moisture.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [Moisture](#)

### 5.18.1 Detailed Description

Header file of [Moisture.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

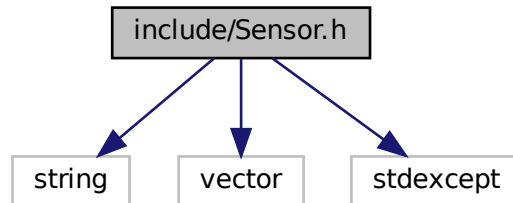
Copyright (c) 2022

## 5.19 include/Sensor.h File Reference

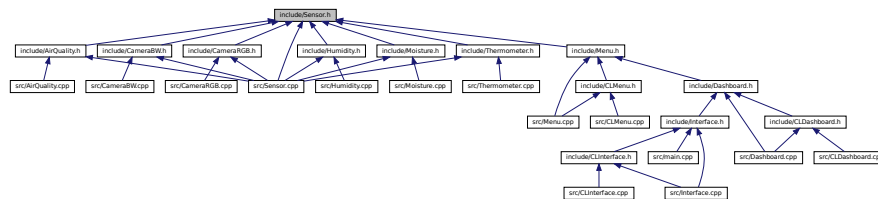
Header file of [Sensor.cpp](#).

```
#include <string>
#include <vector>
#include <stdexcept>
```

Include dependency graph for Sensor.h:



This graph shows which files directly or indirectly include this file:



### Classes

- class [Sensor](#)

### 5.19.1 Detailed Description

Header file of [Sensor.cpp](#).

Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

Version

1.0

Date

2022-11-23

Copyright

Copyright (c) 2022

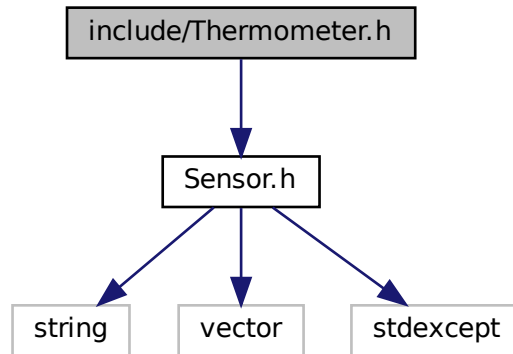


## 5.20 include/Thermometer.h File Reference

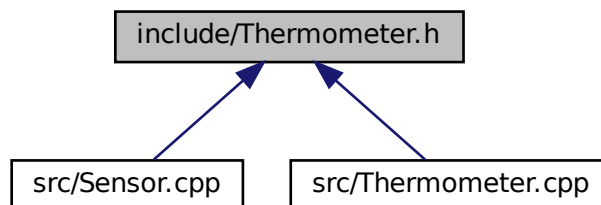
Header file of [Thermometer.cpp](#).

```
#include "Sensor.h"
```

Include dependency graph for Thermometer.h:



This graph shows which files directly or indirectly include this file:



### Classes

- class [Thermometer](#)

#### 5.20.1 Detailed Description

Header file of [Thermometer.cpp](#).

**Author**

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

**Version**

1.0

**Date**

2022-11-23

**Copyright**

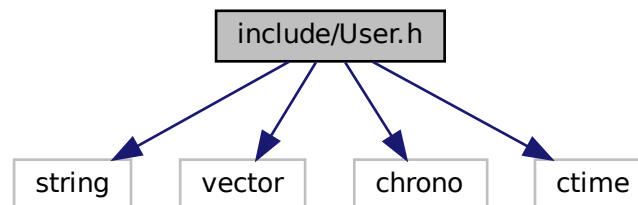
Copyright (c) 2022

## 5.21 include/User.h File Reference

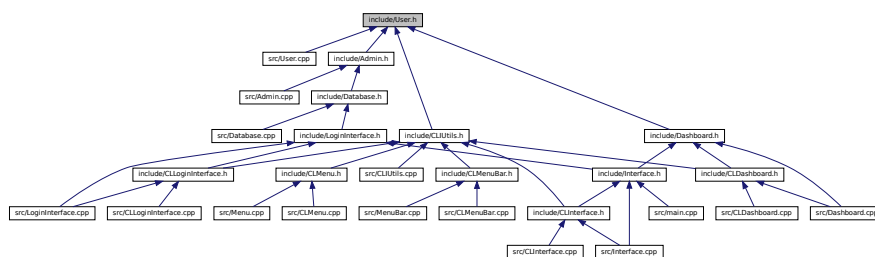
Header file of [User.cpp](#).

```
#include <string>
#include <vector>
#include <chrono>
#include <ctime>
```

Include dependency graph for User.h:



This graph shows which files directly or indirectly include this file:



## Classes

- class [User](#)

### 5.21.1 Detailed Description

Header file of [User.cpp](#).

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

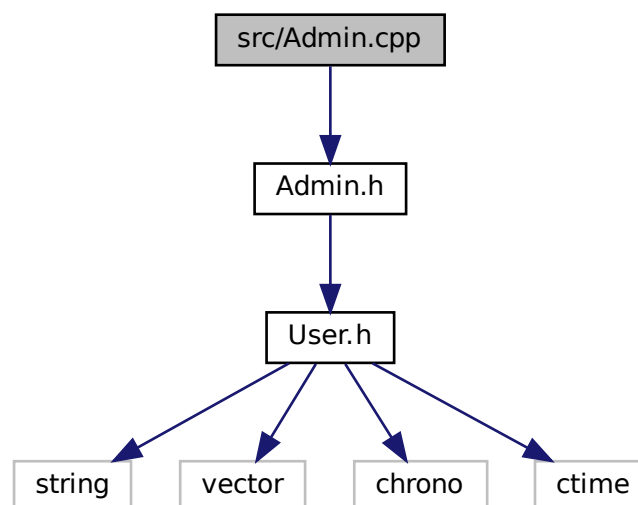
#### Copyright

Copyright (c) 2022

## 5.22 src/Admin.cpp File Reference

```
#include "Admin.h"
```

Include dependency graph for Admin.cpp:



### 5.22.1 Detailed Description

**Author**

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

**Version**

1.0

**Date**

2022-11-23

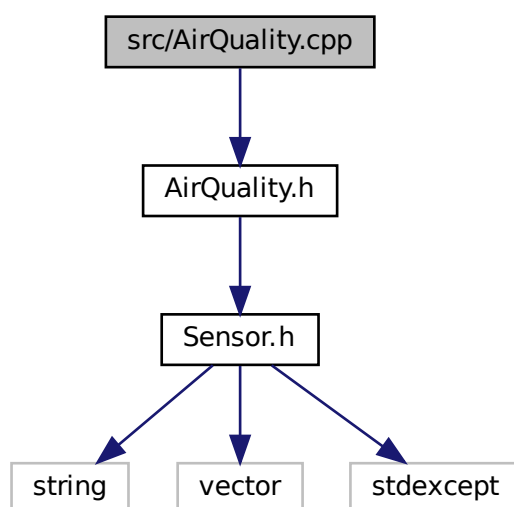
**Copyright**

Copyright (c) 2022

## 5.23 src/AirQuality.cpp File Reference

```
#include "AirQuality.h"
```

Include dependency graph for AirQuality.cpp:



### 5.23.1 Detailed Description

**Author**

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

**Version**

1.0

**Date**

2022-11-23

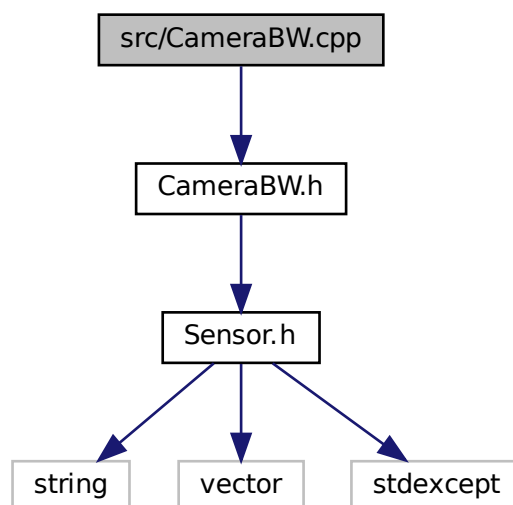
**Copyright**

Copyright (c) 2022

## 5.24 src/CameraBW.cpp File Reference

```
#include "CameraBW.h"
```

Include dependency graph for CameraBW.cpp:



### 5.24.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

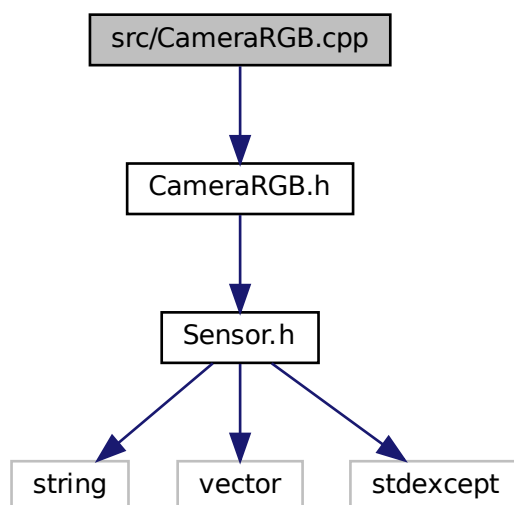
#### Copyright

Copyright (c) 2022

## 5.25 src/CameraRGB.cpp File Reference

```
#include "CameraRGB.h"
```

Include dependency graph for CameraRGB.cpp:



### 5.25.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

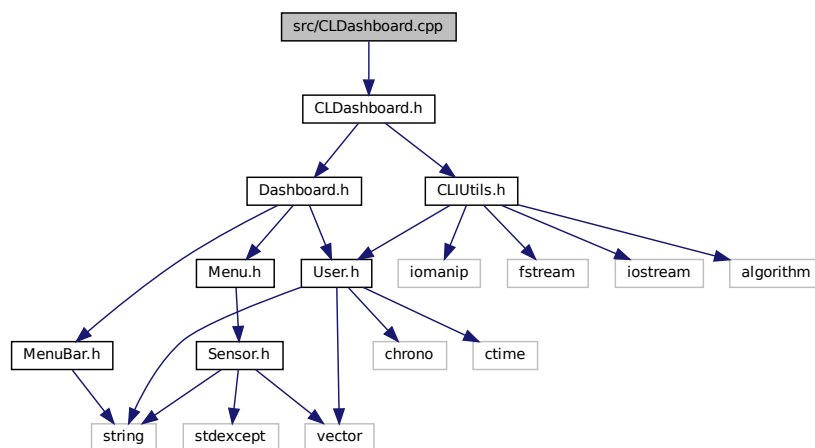
#### Copyright

Copyright (c) 2022

## 5.26 src/CLDashboard.cpp File Reference

```
#include "CLDashboard.h"
```

Include dependency graph for CLDashboard.cpp:



### 5.26.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

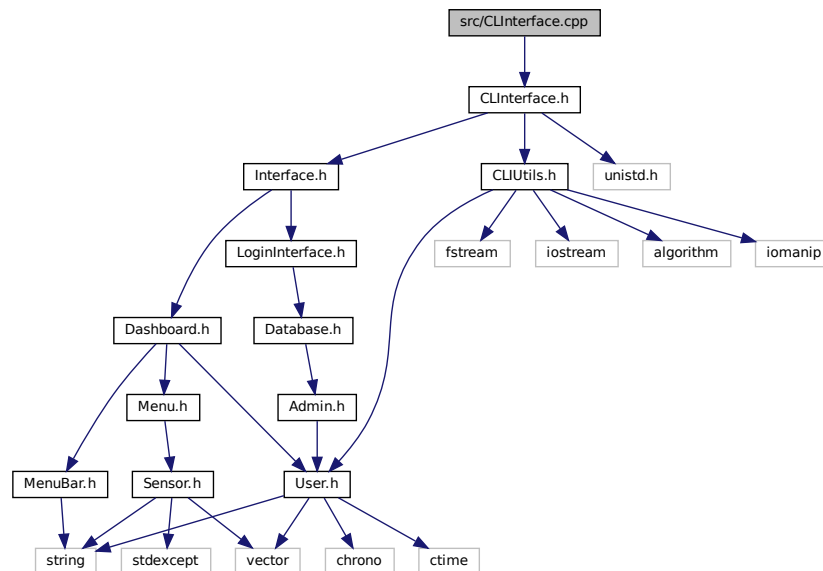
#### Copyright

Copyright (c) 2022

## 5.27 src/CLIInterface.cpp File Reference

```
#include "CLIInterface.h"
```

Include dependency graph for CLIInterface.cpp:



### 5.27.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

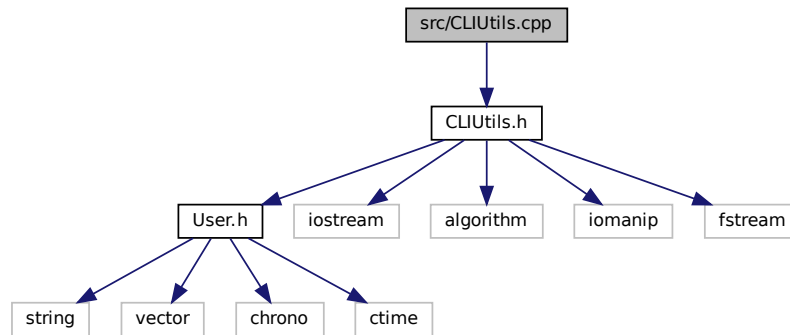
Copyright (c) 2022



## 5.28 src/CLIUtils.cpp File Reference

```
#include "CLIUtils.h"
```

Include dependency graph for CLIUtils.cpp:



### Functions

- void **setTerminalSize** ()
- int **getTerminalWidth** ()
- int **getTerminalHeight** ()
- void **printCenterFromFile** (std::string fileName, std::string color)  
*Print the contents of the file in the center.*
- void **printCenter** (const std::string toPrint, const int padding)  
*Print the string in the center.*
- void **printCenter** (const std::string toPrint, const std::string color, const int padding)  
*Print the string in the center.*
- void **printRight** (const std::string toPrint, const int padding)
- void **printRight** (const std::string toPrint, const std::string color, const int padding)
- void **printLeft** (const std::string toPrint, const int padding)
- void **printLeft** (const std::string toPrint, const std::string color, const int padding)
- void **printColor** (std::string toPrint, std::string color)  
*Print the string to the terminal in the given color.*
- std::string **setColor** (std::string color)  
*Set the Color to print.*
- void **startCustomTerminal** (int terminalSize)
- void **clearCustomTerminal** (int terminalSize)
- void **moveCursor** (int posX, int posY)
- std::vector< std::string > **newCommand** (User &user, std::string currentSensor)
- void **printGraphic** (const std::vector< int > &data, int valPerY, int posX, int posY, int scale)

### Variables

- int **terminalWidth**
- int **terminalHeight**

### 5.28.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

Copyright (c) 2022

### 5.28.2 Function Documentation

#### 5.28.2.1 `printCenter()` [1/2]

```
void printCenter (
    const std::string toPrint,
    const int padding )
```

Print the string in the center.

#### Parameters

<i>toPrint</i>	
<i>padding</i>	

#### 5.28.2.2 `printCenter()` [2/2]

```
void printCenter (
    const std::string toPrint,
    const std::string color,
    const int padding )
```

Print the string in the center.

## Parameters

<i>toPrint</i>	
<i>padding</i>	
<i>color</i>	

**5.28.2.3 printCenterFromFile()**

```
void printCenterFromFile (
    std::string fileName,
    std::string color )
```

Print the contents of the file in the center.

## Parameters

<i>fileName</i>	
<i>color</i>	

**5.28.2.4 printColor()**

```
void printColor (
    std::string toPrint,
    std::string color )
```

Print the string to the terminal in the given color.

## Parameters

<i>toPrint</i>	
<i>color</i>	

**5.28.2.5 setColor()**

```
std::string setColor (
    std::string color )
```

Set the Color to print.

## Parameters

<i>color</i>	
--------------	--

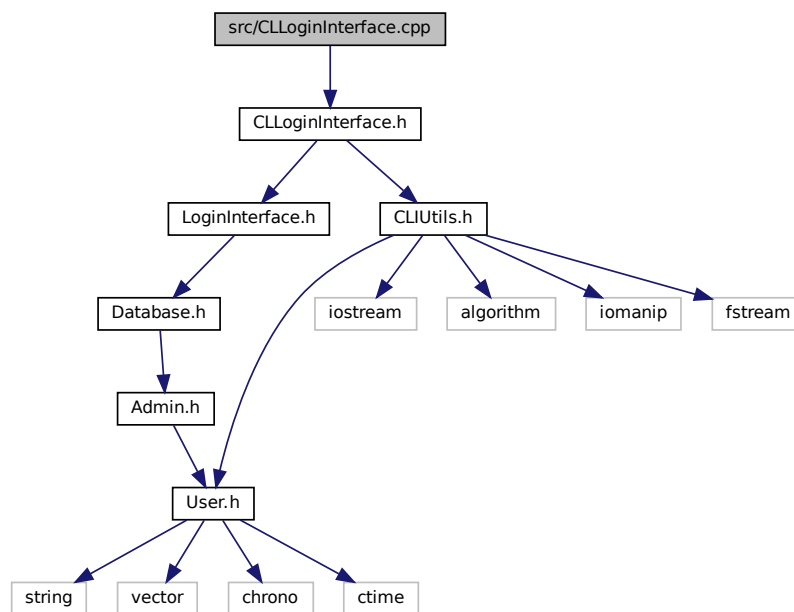
**Returns**

std::string

## 5.29 src/CLLoginInterface.cpp File Reference

```
#include "CLLoginInterface.h"
```

Include dependency graph for CLLoginInterface.cpp:



### 5.29.1 Detailed Description

**Author**

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

**Version**

1.0

**Date**

2022-11-23

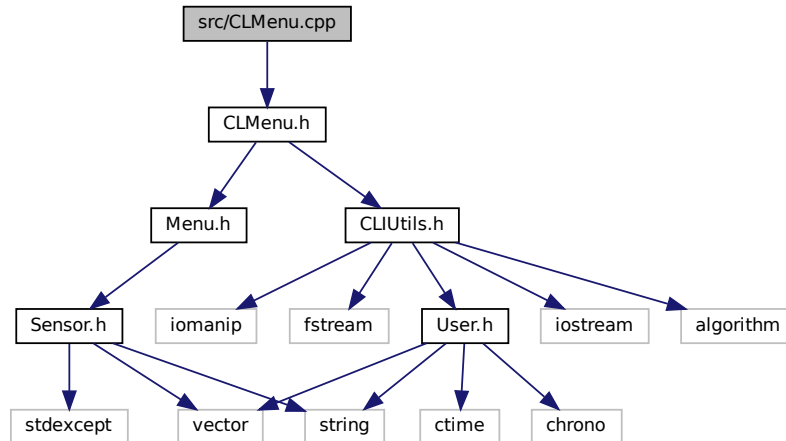
**Copyright**

Copyright (c) 2022

## 5.30 src/CLMenu.cpp File Reference

```
#include "CLMenu.h"
```

Include dependency graph for CLMenu.cpp:



### 5.30.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

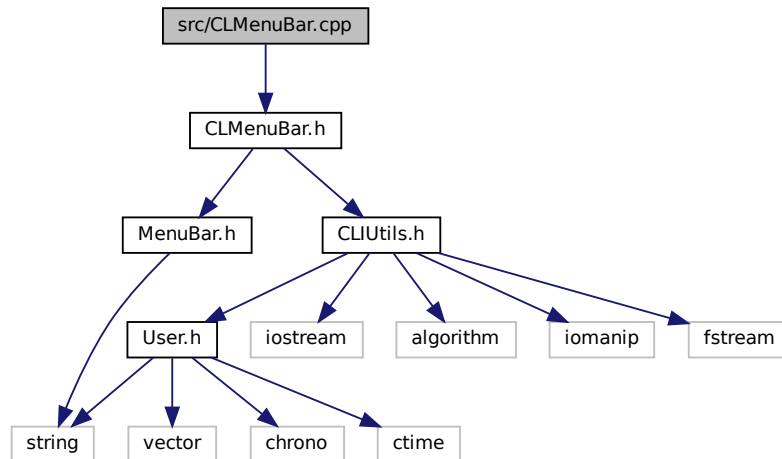
#### Copyright

Copyright (c) 2022

## 5.31 src/CLMenuBar.cpp File Reference

```
#include "CLMenuBar.h"
```

Include dependency graph for CLMenuBar.cpp:



### 5.31.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

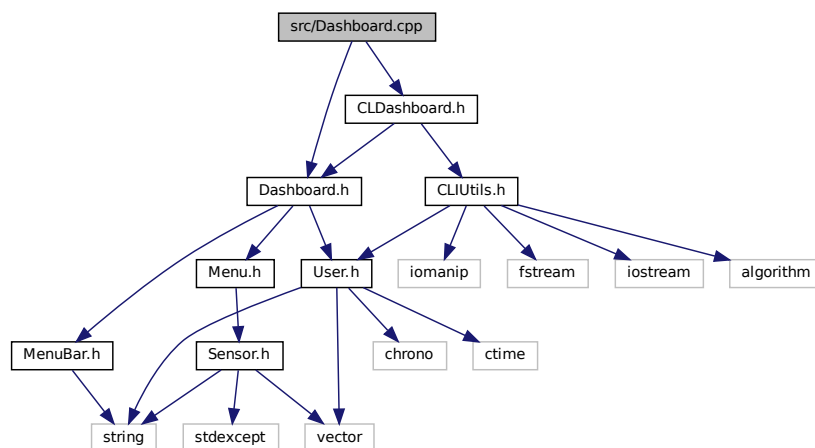
Copyright (c) 2022

## 5.32 src/Dashboard.cpp File Reference

```
#include "Dashboard.h"
```

```
#include "CLDashboard.h"
```

Include dependency graph for Dashboard.cpp:



### 5.32.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

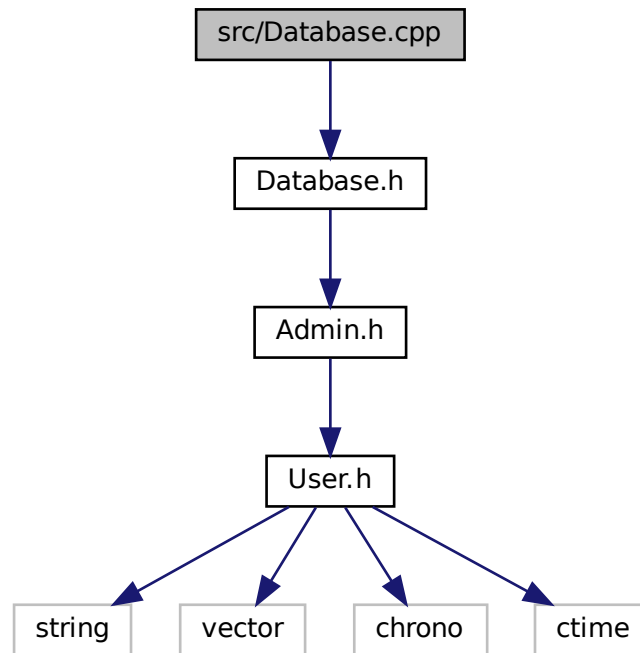
#### Copyright

Copyright (c) 2022

### 5.33 src/Database.cpp File Reference

```
#include "Database.h"
```

Include dependency graph for Database.cpp:



#### 5.33.1 Detailed Description

##### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

##### Version

1.0

##### Date

2022-11-23

##### Copyright

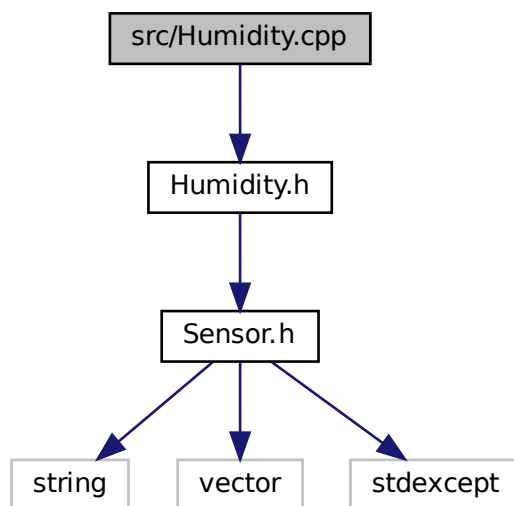
Copyright (c) 2022



## 5.34 src/Humidity.cpp File Reference

```
#include "Humidity.h"
```

Include dependency graph for Humidity.cpp:



### 5.34.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

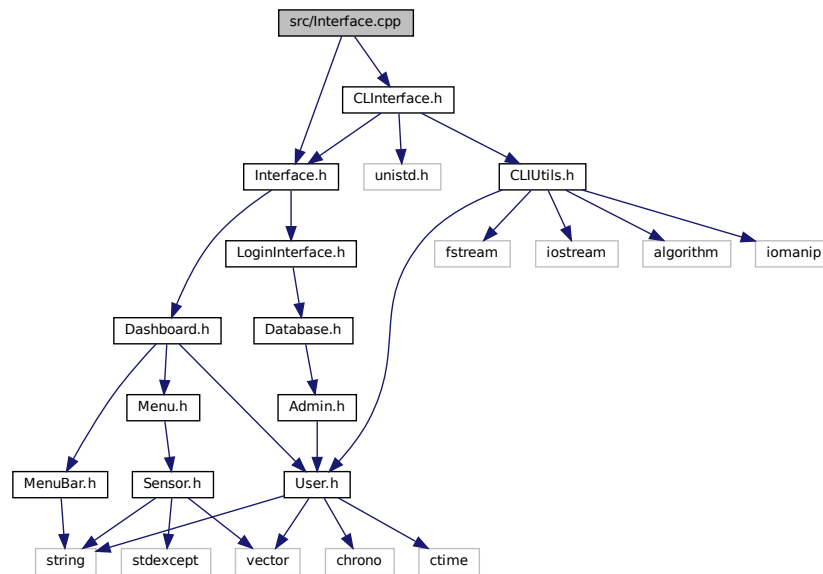
2022-11-23

#### Copyright

Copyright (c) 2022

## 5.35 src/Interface.cpp File Reference

```
#include "Interface.h"
#include "CLIInterface.h"
Include dependency graph for Interface.cpp:
```



### 5.35.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

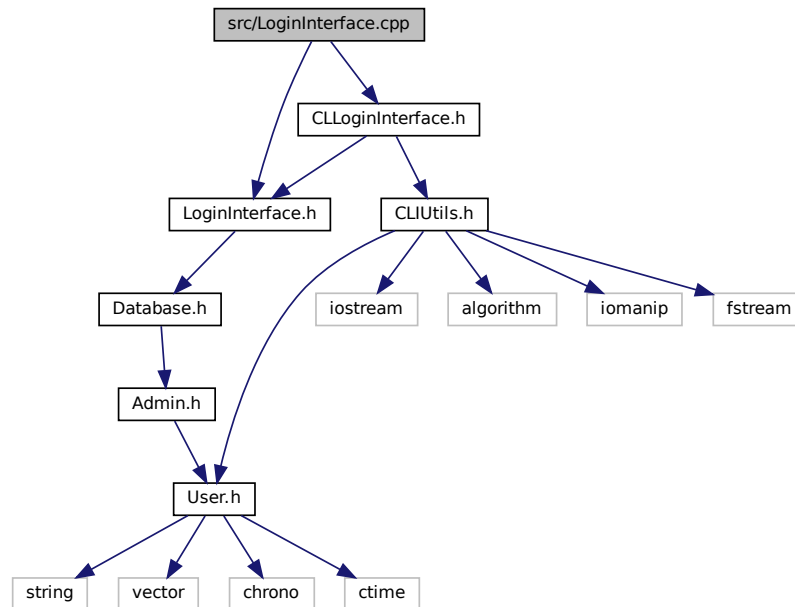
2022-11-23

#### Copyright

Copyright (c) 2022

## 5.36 src/LoginInterface.cpp File Reference

```
#include "LoginInterface.h"
#include "CLLoginInterface.h"
Include dependency graph for LoginInterface.cpp:
```



### 5.36.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

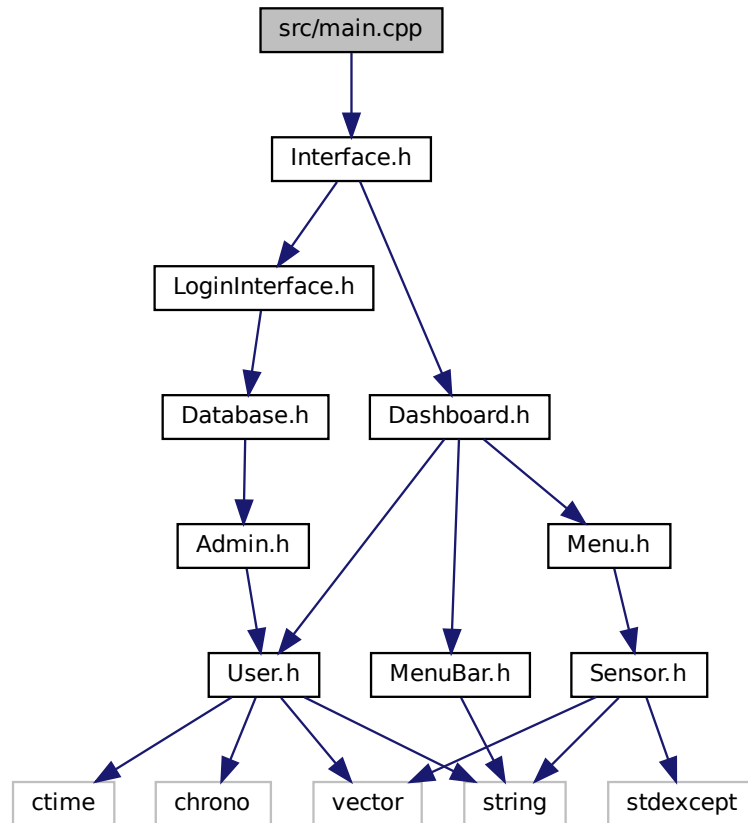
#### Copyright

Copyright (c) 2022

## 5.37 src/main.cpp File Reference

```
#include "Interface.h"
```

Include dependency graph for main.cpp:



### Functions

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

#### 5.37.1 Detailed Description

##### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

##### Version

1.0

## Date

2022-11-23

## Copyright

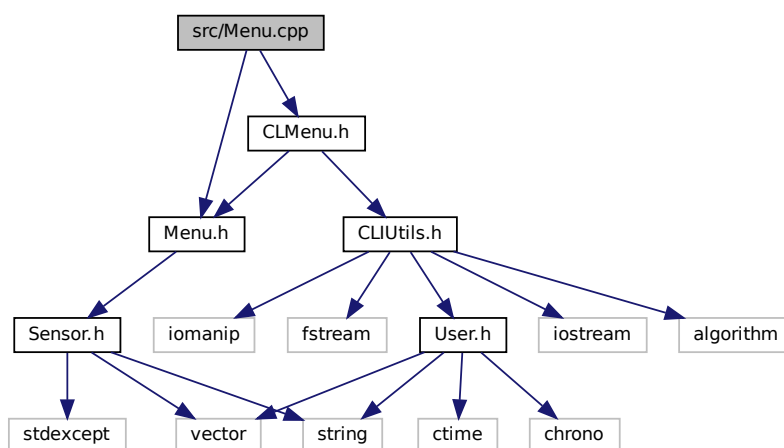
Copyright (c) 2022

## 5.38 src/Menu.cpp File Reference

```
#include "Menu.h"
```

```
#include "CLMenu.h"
```

Include dependency graph for Menu.cpp:



### 5.38.1 Detailed Description

## Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

## Version

1.0

## Date

2022-11-23

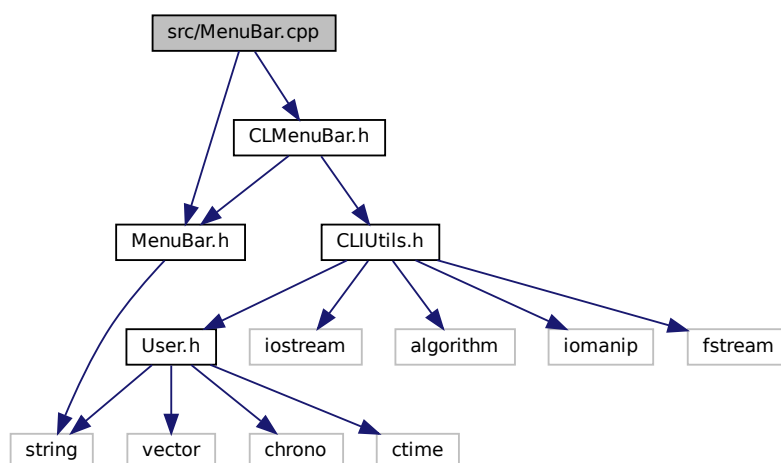
## Copyright

Copyright (c) 2022

## 5.39 src/MenuBar.cpp File Reference

```
#include "MenuBar.h"  
#include "CLMenuBar.h"
```

Include dependency graph for MenuBar.cpp:



### 5.39.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

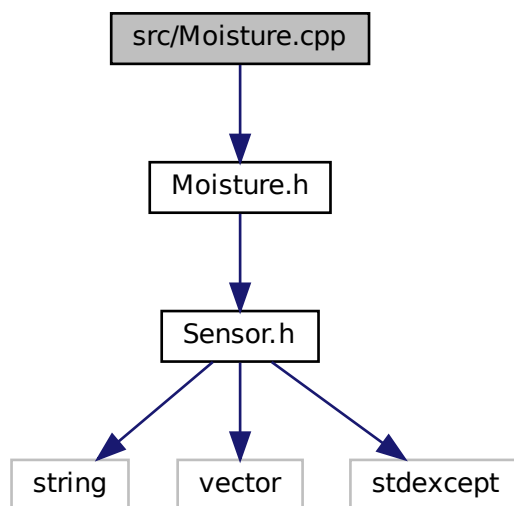
#### Copyright

Copyright (c) 2022

## 5.40 src/Moisture.cpp File Reference

```
#include "Moisture.h"
```

Include dependency graph for Moisture.cpp:



### 5.40.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

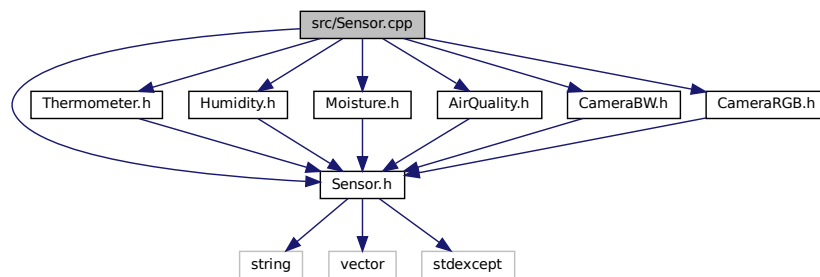
#### Copyright

Copyright (c) 2022

## 5.41 src/Sensor.cpp File Reference

```
#include "Sensor.h"  
#include "Thermometer.h"  
#include "Humidity.h"  
#include "Moisture.h"  
#include "AirQuality.h"  
#include "CameraBW.h"  
#include "CameraRGB.h"
```

Include dependency graph for Sensor.cpp:



### 5.41.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

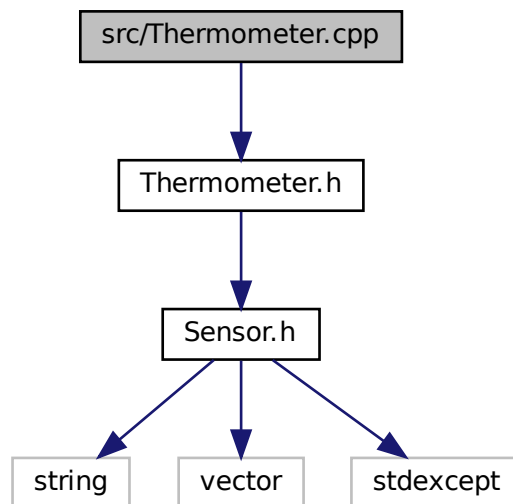
Copyright (c) 2022



## 5.42 src/Thermometer.cpp File Reference

```
#include "Thermometer.h"
```

Include dependency graph for Thermometer.cpp:



### 5.42.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

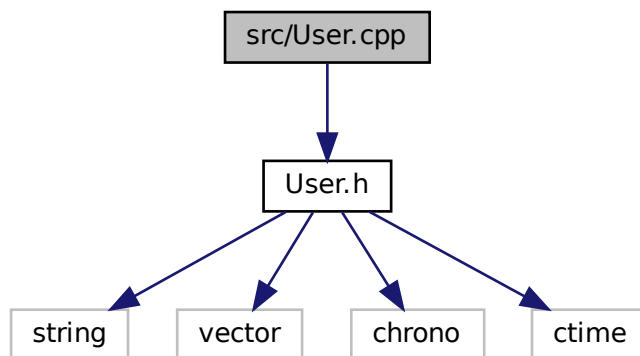
#### Copyright

Copyright (c) 2022

## 5.43 src/User.cpp File Reference

```
#include "User.h"
```

Include dependency graph for User.cpp:



### 5.43.1 Detailed Description

#### Author

Javier Izquierdo ( [j.izquierdoh.2021@alumnos.urjc.es](mailto:j.izquierdoh.2021@alumnos.urjc.es) )

#### Version

1.0

#### Date

2022-11-23

#### Copyright

Copyright (c) 2022

# Index

- addNewSensor
  - Dashboard, [26](#)
- addTimestamp
  - User, [49](#)
- addToMainMenu
  - Dashboard, [27](#)
- Admin, [7](#)
  - Admin, [8](#)
- AirQuality, [8](#)
  - AirQuality, [9](#)
  - requestData, [9](#)
- askEmployeeNumber
  - CLLoginInterface, [20](#)
- askNIF
  - CLLoginInterface, [20](#)
- CameraBW, [10](#)
  - CameraBW, [11](#)
  - requestData, [11](#)
- CameraRGB, [12](#)
  - CameraRGB, [12](#)
  - requestData, [13](#)
- canExit
  - Dashboard, [27](#)
- changeCurrentSensorInfo
  - Dashboard, [27](#)
- changeInterface
  - Dashboard, [28](#)
- changeMainMenu
  - CLDashboard, [15](#)
- checkUser
  - LoginInterface, [36](#)
- CLDashboard, [13](#)
  - changeMainMenu, [15](#)
  - CLDashboard, [14](#)
  - errorCommand, [15](#)
  - helpCommand, [15](#)
  - listSensor, [16](#)
  - permissionError, [16](#)
  - readCommand, [16](#)
  - showMainMenu, [16](#)
- cleanSensor
  - Dashboard, [28](#)
- clearMenu
  - CLMenu, [22](#)
- CLInterface, [17](#)
  - CLInterface, [17](#)
  - login, [18](#)
- CLIUtils.cpp
  - printCenter, [90](#)
  - printCenterFromFile, [91](#)
  - printColor, [91](#)
  - setColor, [91](#)
- CLIUtils.h
  - printCenter, [62, 63](#)
  - printCenterFromFile, [63](#)
  - printColor, [63](#)
  - setColor, [64](#)
- CLLoginInterface, [19](#)
  - askEmployeeNumber, [20](#)
  - askNIF, [20](#)
  - showWelcomeMessage, [21](#)
- CLMenu, [21](#)
  - clearMenu, [22](#)
  - show, [22, 23](#)
- CLMenuBar, [23](#)
  - show, [24](#)
- Create
  - Dashboard, [28](#)
  - Interface, [33](#)
  - LoginInterface, [36](#)
  - Menu, [37](#)
  - MenuBar, [39](#)
  - Sensor, [43](#)
- Dashboard, [25](#)
  - addNewSensor, [26](#)
  - addToMainMenu, [27](#)
  - canExit, [27](#)
  - changeCurrentSensorInfo, [27](#)
  - changeInterface, [28](#)
  - cleanSensor, [28](#)
  - Create, [28](#)
  - deleteSensor, [29](#)
  - exit, [29](#)
  - logout, [29](#)
  - moveWindowMainMenu, [29](#)
  - setUser, [30](#)
- Database, [30](#)
  - deleteSensor
    - Dashboard, [29](#)
- errorCommand
  - CLDashboard, [15](#)
- exit
  - Dashboard, [29](#)
- getArea
  - Sensor, [43](#)
- getId

- Sensor, 43
- getMagnitude
  - Sensor, 44
- getName
  - User, 49
- getTimestamp
  - User, 49
- getType
  - Sensor, 44
- getUser
  - LoginInterface, 36
- getValPerMin
  - Sensor, 44
- hasAdminPermission
  - User, 50
- helpCommand
  - CLDashboard, 15
- Humidity, 31
  - Humidity, 31
  - requestData, 32
- include/Admin.h, 53
- include/AirQuality.h, 55
- include/CameraBW.h, 56
- include/CameraRGB.h, 57
- include/CLDashboard.h, 59
- include/CLInterface.h, 60
- include/CLIUtils.h, 61
- include/CLLoginInterface.h, 64
- include/CLMenu.h, 66
- include/CLMenuBar.h, 67
- include/Dashboard.h, 68
- include/Database.h, 70
- include/Humidity.h, 71
- include/Interface.h, 73
- include/LoginInterface.h, 74
- include/Menu.h, 76
- include/MenuBar.h, 77
- include/Moisture.h, 78
- include/Sensor.h, 80
- include/Thermometer.h, 81
- include/User.h, 82
- Interface, 32
  - Create, 33
  - loadMenu, 34
  - login, 34
- isActive
  - Sensor, 44
- isSameEmployeeNumber
  - User, 50
- isSameNIF
  - User, 50
- listSensor
  - CLDashboard, 16
- loadMenu
  - Interface, 34
- login
  - CLInterface, 18
  - Interface, 34
  - LoginInterface, 35
    - checkUser, 36
    - Create, 36
    - getUser, 36
  - logout
    - Dashboard, 29
- Menu, 37
  - Create, 37
- MenuBar, 38
  - Create, 39
  - setCurrentMenu, 39
  - setUserName, 39
- Moisture, 40
  - Moisture, 40
  - requestData, 41
- moveWindowMainMenu
  - Dashboard, 29
- permissionError
  - CLDashboard, 16
- printCenter
  - CLIUtils.cpp, 90
  - CLIUtils.h, 62, 63
- printCenterFromFile
  - CLIUtils.cpp, 91
  - CLIUtils.h, 63
- printColor
  - CLIUtils.cpp, 91
  - CLIUtils.h, 63
- readCommand
  - CLDashboard, 16
- requestData
  - AirQuality, 9
  - CameraBW, 11
  - CameraRGB, 13
  - Humidity, 32
  - Moisture, 41
  - Thermometer, 47
- Sensor, 41
  - Create, 43
  - getArea, 43
  - getId, 43
  - getMagnitude, 44
  - getType, 44
  - getValPerMin, 44
  - isActive, 44
  - Sensor, 42
  - setActive, 45
  - setArea, 45
  - setMagnitude, 45
  - setValPerMin, 46
- setActive
  - Sensor, 45
- setArea

- Sensor, [45](#)
- setColor
  - CLIUtils.cpp, [91](#)
  - CLIUtils.h, [64](#)
- setCurrentMenu
  - MenuBar, [39](#)
- setEmployeeNumber
  - User, [51](#)
- setMagnitude
  - Sensor, [45](#)
- setNIF
  - User, [51](#)
- setUser
  - Dashboard, [30](#)
- setUserName
  - MenuBar, [39](#)
- setValPerMin
  - Sensor, [46](#)
- show
  - CLMenu, [22](#), [23](#)
  - CLMenuBar, [24](#)
- showMainMenu
  - CLDashboard, [16](#)
- showWelcomeMessage
  - CLLoginInterface, [21](#)
- src/Admin.cpp, [83](#)
- src/AirQuality.cpp, [84](#)
- src/CameraBW.cpp, [85](#)
- src/CameraRGB.cpp, [86](#)
- src/CLDashboard.cpp, [87](#)
- src/CLInterface.cpp, [88](#)
- src/CLIUtils.cpp, [89](#)
- src/CLLoginInterface.cpp, [92](#)
- src/CLMenu.cpp, [93](#)
- src/CLMenuBar.cpp, [94](#)
- src/Dashboard.cpp, [95](#)
- src/Database.cpp, [96](#)
- src/Humidity.cpp, [97](#)
- src/Interface.cpp, [98](#)
- src/LoginInterface.cpp, [99](#)
- src/main.cpp, [100](#)
- src/Menu.cpp, [101](#)
- src/MenuBar.cpp, [102](#)
- src/Moisture.cpp, [103](#)
- src/Sensor.cpp, [104](#)
- src/Thermometer.cpp, [105](#)
- src/User.cpp, [106](#)
- Thermometer, [46](#)
  - requestData, [47](#)
  - Thermometer, [47](#)
- User, [48](#)
  - addTimestamp, [49](#)
  - getName, [49](#)
  - getTimestamp, [49](#)
  - hasAdminPermission, [50](#)
  - isSameEmployeeNumber, [50](#)
  - isSameNIF, [50](#)
  - setEmployeeNumber, [51](#)
  - setNIF, [51](#)
  - User, [49](#)