



TCP LUMINOSITE

```
- sendData(tcp::socket& socket) : bool
```

- endpoint_ : tcp::endpoint
- maxQueueSize = 3 : static const size t
- dataQueue : std::deque<std::string>
- lever : bool

+ TCP_LUMINOSITE(const std::string& serverAddress, int serverPort)

- + connectAndSend() : void
- + createDateTime() : std::string
- + startServer() : std::string
- + getLever() : bool
- + setLeverTrue() : void
- + setLeverFalse() : void

ApplicationWeb

Consommation

```
- sendData(tcp::socket& socket) : bool
```

- endpoint_ : tcp::endpoint
- maxQueueSize = 3 : static const size
- dataQueue_ : std::deque<std::string
- + LUMINOSITE(const std::string& serve
- + connectAndSend() : void
- + crostoDatoTime() + std. string

Lum



inosite

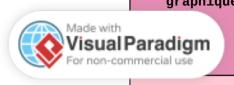
e_t

Address, int serverPort)

ARDUINO CARTE E.S

- sendData(tcp::socket& socket) : bool
- endpoint_ : tcp::endpoint
- maxQueueSize = 3 : static const size_t
- dataQueue_ : std::deque<std::string>
- io_context_ : boost::asio::io_context
- serialPort_ : boost::asio::serial_port
- buffer : std::string
- receivedData_ : std::string
- + ArduinoCarteES(const std::string& serverAddress, int serverPort, const std::st serialPort)
- + connectAndSend() + void





	_	_	м								,
a	f	f	1	C	h	e	r	Ι	Н	М	(
											•

SimulationAPI									
donneesSimulees: DonneeConsommation[] DonneePanneauSolaire[]	7								
simulerDonnees(): void									

DonneeConsomm chargerDonnee afficherGraph

donneesConsor

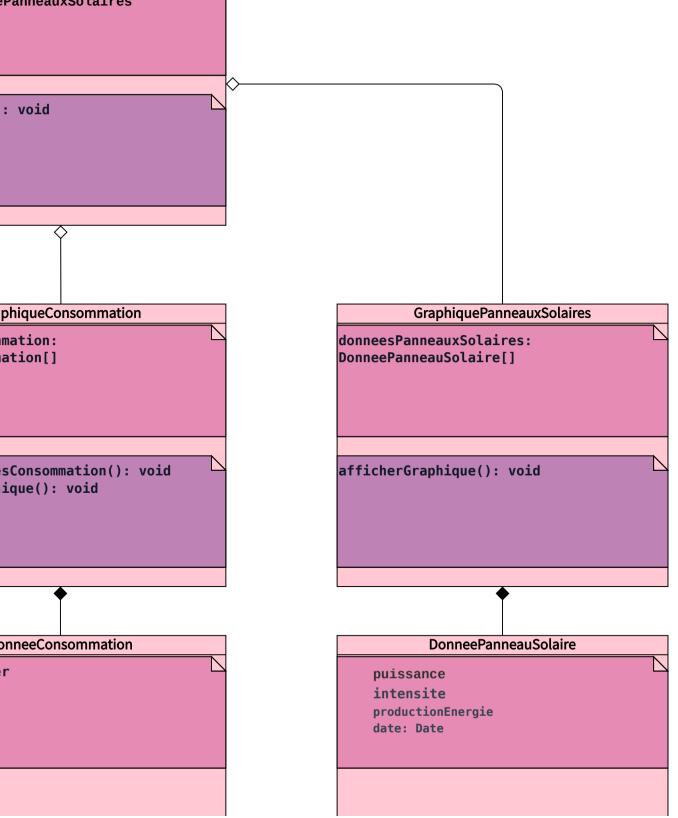
Gra

valeur: numbe

EnergieRecordAPI

-create(record)

-read(id)





- sendData(tcp::socket& socket) : boo

- endpoint_ : tcp::endpoint - maxQueueSize = 3 : static const size

+ CarteES(const std::string& serverAdo

- maxQueueSize = 3 : static const size dataQueue_ : std::deque<std::string
- + connectAndSend() : void
 + createDateTime() : std::string

Con

- proportionsTempVertStockees : array [asourceVerteStockees : array [int]
- MAX_SIZE = 10 : int - previousSourceVerte : int
- previoussourceverte : int - previoustabPowerBox : array [int] - confia : string
- dataAPIBox : array [json object]dataAPIAcces : array [json object]

dataAPISolarPannel : array [json objecdataAPILumi : array [json object]

- MAX_SIZE_API : int - reconnectionIntervals : int
- firstTimer : Date
 timer : Date
- previousTimer : DatetimerVert : int
- timerTotal : intMoyennetempsPuissanceBox : array [floa
- tempPuissanceBox : array [int]
- ratio : float
- + loadConfig() : json object
- + async ecouterDonneesCpp() : Promise<vo

+ asvnc SendBoxDataToAPI(string config.

```
For non-commercial use
rteES
                                                                                      ARDUINO LUMINOSITE
e t
                                                  - sendData(tcp::socket& socket) : bool
                                                  - endpoint : tcp::endpoint
                                                  - maxQueueSize = 3 : static const size t
dress, int serverPort)
                                                  - dataQueue : std::deque<std::string>
                                                  - io context : boost::asio::io context
                                                  - serialPort : boost::asio::serial port
                                                  - buffer : std::string
                                                  - receivedData : std::string
noteurs
                                                  + ARDUINO LUMINOSITE(const std::string& serverAddress, int serverPort, const std
                                                 serialPort)
                                                  + connectAndSend() : void
rrav[float]]
                                                  + createDateTime() : std::string
                                                  + connectSerial(const std::string& port, unsigned int baud rate=9600) : void
                                                  + readFromSerial() : std::string
                                                                                    Capteur luminosite
                                                  - m carte : int
                                                  - numcarte : int
                                                  - sendData(tcp::socket& socket) : bool
                                                  - serverAddress : std::string
                                                  - serverPort : int
                                                  - maxQueueSize = 3 : static const size t
                                                  - dataQueue_ : std::deque<std::string>
                                                  - endpoint : sockaddr in
                                                  - resistance = 44.6 : double
                                                  + capteur luminosite(const std::string& serverAddress, int serverPort, int carte
                                                 PCI 9111DG)
                                                  + ~capteur luminosite()
                                                  + lire tension(double& tension, int canal = 0, int gamme = AD B 10 V) : void
id>
                                                  + lire tension(int canal = 0, int gamme = AD B 10 V) : double
arrav data[]. string dataKev)
                                                  + lire tension AT ReadChannel (int canal = 0 int damme = AD R 10 V) · unsigned s
```

T COMMECTAMOSEMON : VOIG

+ createDateTime() : std::string

+ readFromSerial() : std::string

+ connectSerial(const std::string& port, unsigned in

Made with

Nisual Paradigm



::string&

hort

- -update(record)
- -delete(id)



: Promise<void>
+ calculerPrc
+ async reco
Promise<number
- async queue(object data, array reach,
Promise<number>
+ async boucle(json object dataCPP) : Pr

```
int]) : json object
array reach, string dataKey) :
string configKey, string dataKey) :
```

omise<void>

+ getnumcarte() : int

+ setnumcarte() : void + connectAndSend() : void

+ createDateTime() : std::string

+ getresistance() : double



