# Aqua Control

1.0

Generated by Doxygen 1.8.17

# Chapter 1

# Todo List

**File main.cpp**

Implementation List

- Turn on Led
- OnOff Pump

# Chapter 2

# Bug List

**File main.cpp**

Bugs list

# Chapter 3

# Class Index

## 3.1 Class List

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

# Chapter 4

# File Index

## 4.1 File List

Here is a list of all files with brief descriptions:

# Chapter 5

# Class Documentation

## 5.1 httpUpdater Class Reference

Update class.

### 5.1.1 Detailed Description

Update class.

provide update for firmware on the http request

**See also**

other classes

- serverA
- RGB

## 5.2 RGB Struct Reference

Struct for Led.

### Public Attributes

- byte r

  *RECEIVE BYTE COLOR RED.*
- byte g

  *RECEIVE BYTE COLOR GREEN.*
- byte b

### 5.2.1 Detailed Description

Struct for Led.

Generates struct for rgb color object

**See also**

other classes

- serverA
- httpUpdater

Definition at line 93 of file main.cpp.

### 5.2.2 Member Data Documentation

#### 5.2.2.1 b

`byte RGB::b`

RECEIVE BYTE COLOR BLUE

Definition at line 100 of file main.cpp.

#### 5.2.2.2 g

`byte RGB::g`

RECEIVE BYTE COLOR GREEN.

Definition at line 98 of file main.cpp.

#### 5.2.2.3 r

`byte RGB::r`

RECEIVE BYTE COLOR RED.

Definition at line 96 of file main.cpp.

## 5.3 serverA Class Reference

WebServer Class.

### 5.3.1 Detailed Description

WebServer Class.

This create a object for handle request response of client

**Parameters**

| | |
|---|---|
| *80* | - Web port |

**See also**

other classes

- httpUpdater
- RGB

# Chapter 6

# File Documentation

## 6.1 main.cpp File Reference

Controls pool pump and RGB led.

```
#include <Arduino.h>
#include <ESP8266WiFi.h>
#include <ESP8266WebServer.h>
#include <FS.h>
#include <LittleFS.h>
#include <ESP8266mDNS.h>
#include <ESP8266HTTPUpdateServer.h>
```

### Classes

- struct RGB

  *Struct for Led.*

### Functions

- ESP8266WebServer serverA (80)
- void handleRoot ()

  *manipulates index.htm file*
- void handleCss ()

  *Set stylesheet of page.*
- void handleJs ()

  *set script of page*
- void handleAction ()

  *function to do action requested by client*
- void handleNotFound ()

  *handle error page*
- void led (RGB colorname)

  *Set color led.*
- bool wait ()

  *Wait for set time.*
- String readFile (String path)

  *Read File.*
- void setup ()

  *Setting source.*
- void loop ()

### Variables

- const char * user = "admin"
- const char * pass = "senha123"
- const char * host = "aqua"
- const char * update_path = "/firmware"
- const char * SSID = "YOURSSID"

  *Ssid Network.*
- const char * PASSWD = "YOURPASSWORD"

  *Password for Network.*
- bool hand = true
- unsigned long setTime = 1000
- unsigned long previusTime = 0
- File uploadFile
- int pinR = 12
- int pinG = 13
- int pinB = 14
- int pump = 5

  *define pin of pump*
- int sunLight = 4
- String page = ""

  *define ldr sensor*
- ESP8266HTTPUpdateServer httpUpdater

  *INSTANCE FOR HTTP UPDATE.*
- RGB color = {0,0,0}

#### 6.1.1 Detailed Description

Controls pool pump and RGB led.

**Author**

Cristiano Rocha

**Warning**

This source is only first implementation

**Todo** Implementation List

- Turn on Led
- OnOff Pump

**Bug** Bugs list

**Copyright**

### 6.1.2 Function Documentation

#### 6.1.2.1 handleAction()

```
void handleAction ( )
```

function to do action requested by client

handles customer request

```
turn on led, turn on pump and all actions
```

**Parameters**

| | |
|---|---|
| - | void |

**Returns**

void

On Off Pump

Definition at line 351 of file main.cpp.

#### 6.1.2.2 handleCss()

```
void handleCss ( )
```

Set stylesheet of page.

set style page

```
set style on page with file.css
```

**Parameters**

| | |
|---|---|
| - | void parameters |

**Returns**

void

Definition at line 330 of file main.cpp.

#### 6.1.2.3 handleJs()

```
void handleJs ( )
```

set script of page

Javascript source.

```
   set script on the page whit file.js
```

**Parameters**

| | |
|---|---|
| - | void parameters |

**Returns**

void

Definition at line 337 of file main.cpp.

#### 6.1.2.4 handleNotFound()

```
void handleNotFound ( )
```

handle error page

handle error 404

```
   response for request error
```

**Parameters**

| | |
|---|---|
| - | void |

**Returns**

void

Definition at line 344 of file main.cpp.

#### 6.1.2.5 handleRoot()

```
void handleRoot ( )
```

manipulates index.htm file

```
send response a first requisition
```

**Parameters**

| | |
|---|---|
| - | void parameters |

**Returns**

void

Definition at line 323 of file main.cpp.

#### 6.1.2.6 led()

```
void led (
            RGB colorname )
```

Set color led.

turn on led in the desired color

```
receive color for turn on led
```

**Parameters**

| | |
|---|---|
| *colorname* | - struct type {r,g,b} |

**Returns**

void

Definition at line 300 of file main.cpp.

#### 6.1.2.7 loop()

```
void loop ( )
```

Definition at line 258 of file main.cpp.

#### 6.1.2.8 readFile()

```
String readFile (
            String path )
```

Read File.

Read file SPIFFS.

```
Open and read all files in the flash system
```

#### 6.1.2.8.1 Variables String *content* -— //return variable
**File** *file* -— //add file data

**Parameters**

| | |
|---|---|
| *path* | - File name |

**Returns**

String read from file

Definition at line 309 of file main.cpp.

### 6.1.2.9 serverA()

```
ESP8266WebServer serverA (
             80  )
```

### 6.1.2.10 setup()

```
setup ( )
```

Setting source.

```
   put all configuration setup of device
```

Definition at line 206 of file main.cpp.

### 6.1.2.11 wait()

```
bool wait ( )
```

Wait for set time.

generates delay without locking the code

```
     set delay with millis() function
```

**Parameters**

| | |
|---|---|
| - | void |

**Returns**

Boolean type

Definition at line 287 of file main.cpp.

### 6.1.3 Variable Documentation

#### 6.1.3.1 color

```
RGB color = {0,0,0}
```

object RGB color

Definition at line 103 of file main.cpp.

#### 6.1.3.2 hand

```
bool hand = true
```

handle manual or automatic action

Definition at line 43 of file main.cpp.

#### 6.1.3.3 host

```
const char* host = "aqua"
```

hostname for local access

Definition at line 39 of file main.cpp.

#### 6.1.3.4 httpUpdater

```
ESP8266HTTPUpdateServer httpUpdater
```

INSTANCE FOR HTTP UPDATE.

Definition at line 68 of file main.cpp.

#### 6.1.3.5 page

```
String page = ""
```

define ldr sensor

Store page HTML

Definition at line 52 of file main.cpp.

#### 6.1.3.6 pass

```
const char* pass = "senha123"
```

paasword for access

Definition at line 38 of file main.cpp.

#### 6.1.3.7 PASSWD

```
const char* PASSWD = "YOURPASSWORD"
```

Password for Network.

Definition at line 42 of file main.cpp.

#### 6.1.3.8 pinB

```
int pinB = 14
```

Set pin blue as number 14

Definition at line 49 of file main.cpp.

#### 6.1.3.9 pinG

```
int pinG = 13
```

Set pin green as number 13

Definition at line 48 of file main.cpp.

#### 6.1.3.10 pinR

```
int pinR = 12
```

Set pin red as number 12

Definition at line 47 of file main.cpp.

#### 6.1.3.11 previusTime

```
unsigned long previusTime = 0
```

receive previus time for trigger delay

Definition at line 45 of file main.cpp.

#### 6.1.3.12 pump

```
int pump = 5
```

define pin of pump

Definition at line 50 of file main.cpp.

#### 6.1.3.13 setTime

```
unsigned long setTime = 1000
```

set time delay

Definition at line 44 of file main.cpp.

#### 6.1.3.14 SSID

```
const char* SSID = "YOURSSID"
```

Ssid Network.

Definition at line 41 of file main.cpp.

#### 6.1.3.15 sunLight

```
int sunLight = 4
```

Definition at line 51 of file main.cpp.

#### 6.1.3.16 update_path

```
const char* update_path = "/firmware"
```

Path for update

Definition at line 40 of file main.cpp.

#### 6.1.3.17 uploadFile

```
File uploadFile
```

Store data for SPIFFS readed

Definition at line 46 of file main.cpp.

#### 6.1.3.18 user

```
const char* user = "admin"
```

username for login

Definition at line 37 of file main.cpp.

# Index