

WesleyCommander

Generated by Doxygen 1.8.4

Tue Mar 18 2014 17:34:54

Contents

| | | |
|----------|--|----------|
| 1 | Class Index | 1 |
| 1.1 | Class List | 1 |
| 2 | Class Documentation | 3 |
| 2.1 | ExitHandler Class Reference | 3 |
| 2.1.1 | Constructor & Destructor Documentation | 3 |
| 2.1.1.1 | ExitHandler | 3 |
| 2.1.2 | Member Function Documentation | 3 |
| 2.1.2.1 | id_flame | 3 |
| 2.1.2.2 | id_tool | 3 |
| 2.2 | LedNotifier Class Reference | 3 |
| 2.2.1 | Constructor & Destructor Documentation | 4 |
| 2.2.1.1 | LedNotifier | 4 |
| 2.2.2 | Member Function Documentation | 4 |
| 2.2.2.1 | parse | 4 |
| 2.2.2.2 | throwGeneralFailure | 4 |
| 2.2.2.3 | throwLedCode | 4 |
| 2.3 | Logger Class Reference | 4 |
| 2.3.1 | Detailed Description | 5 |
| 2.3.2 | Constructor & Destructor Documentation | 5 |
| 2.3.2.1 | Logger | 5 |
| 2.3.3 | Member Function Documentation | 5 |
| 2.3.3.1 | logStatus | 5 |
| | Index | 6 |

Chapter 1

Class Index

1.1 Class List

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

| | | |
|-----------------------------|-------|---|
| ExitHandler | | 3 |
| LedNotifier | | 3 |
| Logger | | 4 |

Chapter 2

Class Documentation

2.1 ExitHandler Class Reference

Public Member Functions

- [ExitHandler](#) ([Logger](#) *logger_)
- void [id_flame](#) (int)
- void [id_tool](#) (int)

2.1.1 Constructor & Destructor Documentation

2.1.1.1 ExitHandler::ExitHandler ([Logger](#) * *logger_*)

Constructs an exithandler class

This class should be used to handle the return codes for wesleys binaries

2.1.2 Member Function Documentation

2.1.2.1 void ExitHandler::id_flame (int *returnCode*)

Handles the ID flame codes 0 - throws [id_flame_notfound](#) led and sets flame to 0 1 - sets flame to 1 and throws [id_flame_square](#) 2 - sets flame to 2 and throws [id_flame_triangle](#) 3 - sets flame to 3 and throws [id_flame_circle](#) otherwise - throws [cant_find_file](#)

2.1.2.2 void ExitHandler::id_tool (int)

Handles the [id_tool](#) binary

0 - throw [id_tool_failure](#) else - set tool to [returnCode](#)

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

- [src/exit_handlers.h](#)
- [src/exit_handlers.cpp](#)

2.2 LedNotifier Class Reference

Public Member Functions

- [LedNotifier](#) (bool parseOnConstruction=false)
- void [parse](#) (const char *parseFileName="notify_id.txt")
- bool [throwLedCode](#) (string code, bool throwGeneralErrorOnFailure=false)
- void [throwGeneralFailure](#) ()

2.2.1 Constructor & Destructor Documentation

2.2.1.1 [LedNotifier::LedNotifier](#) (bool *parseOnConstruction* = false)

Constructor for [LedNotifier](#).

Parameters

| | |
|-----------------------------|---|
| <i>parseOn-Construction</i> | if set to true, will parse default file on construction |
|-----------------------------|---|

2.2.2 Member Function Documentation

2.2.2.1 void [LedNotifier::parse](#) (const char * *parseFileName* = "notify_id.txt")

Parses a text file

The format of this file goes failure_id (Needs to be one word) green led0 green led1 yellow led0 yellow led1 red led0 red led1

in the event of a parse failure the leds will throw a general failure

multiple files can be parsed. If there are overlaps than the most recently parsed code will stay.

Parameters

| | |
|----------------------|------------------------|
| <i>parseFileName</i> | The text file to parse |
|----------------------|------------------------|

2.2.2.2 void [LedNotifier::throwGeneralFailure](#) () [inline]

A function that will light the general failure leds note: doesn't rely on a successful parse

2.2.2.3 bool [LedNotifier::throwLedCode](#) (string *code*, bool *throwGeneralErrorOnFailure* = false)

Throws a new led code based on previously parsed file. if throwGeneralErrorOnFailure is true then if no code is found the function will return false and light the general failure led

Returns

true if code was found

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

- src/notifier.h
- src/notifier.cpp

2.3 Logger Class Reference

```
#include <logger.h>
```


Public Member Functions

- [Logger](#) ()
- void [logStatus](#) (string message)

2.3.1 Detailed Description

Class to log status to status node.

2.3.2 Constructor & Destructor Documentation

2.3.2.1 `Logger::Logger ()`

Constructs the [Logger](#) class

Starts ros publishing node "commanderloggerpub" Publishes to node /master/logger

2.3.3 Member Function Documentation

2.3.3.1 `void Logger::logStatus (string message)`

Writes a message to the logging topic

Message format is: (name of node) (randomly chosen id for this commander session) (message number) message

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

- src/logger.h
- src/logger.cpp

Index

- ExitHandler, [3](#)
 - ExitHandler, [3](#)
 - ExitHandler, [3](#)
 - id_flame, [3](#)
 - id_tool, [3](#)
- id_flame
 - ExitHandler, [3](#)
- id_tool
 - ExitHandler, [3](#)
- LedNotifier, [3](#)
 - LedNotifier, [4](#)
 - LedNotifier, [4](#)
 - parse, [4](#)
 - throwGeneralFailure, [4](#)
 - throwLedCode, [4](#)
- logStatus
 - Logger, [5](#)
- Logger, [4](#)
 - logStatus, [5](#)
 - Logger, [5](#)
- parse
 - LedNotifier, [4](#)
- throwGeneralFailure
 - LedNotifier, [4](#)
- throwLedCode
 - LedNotifier, [4](#)