



QxCppUnit Library Version 0.9.2

# **QxCppUnit Library**

## **Reference Manual**



# Contents

---

<b>1</b>	<b>Main Page</b>	<b>1</b>
1.1	Introduction . . . . .	1
<b>2</b>	<b>License</b>	<b>3</b>
<b>3</b>	<b>Implementation Aspects</b>	<b>5</b>
3.1	Exported Symbols . . . . .	6
3.2	Result Types . . . . .	6
<b>4</b>	<b>Examples</b>	<b>7</b>
4.1	A CppUnit Test Example . . . . .	8
4.2	The Demo Main Program . . . . .	8
<b>5</b>	<b>Directory Documentation</b>	<b>11</b>
5.1	qxrunner/include/ Directory Reference . . . . .	11
5.2	qxrunner/include/qxcppunit/ Directory Reference . . . . .	12
5.3	qxrunner/src/qxcppunit/ Directory Reference . . . . .	13
5.4	qxrunner/ Directory Reference . . . . .	14
5.5	qxrunner/src/ Directory Reference . . . . .	15
<b>6</b>	<b>Namespace Documentation</b>	<b>17</b>
6.1	CPPUNIT_NS Namespace Reference . . . . .	17
6.1.1	Detailed Description . . . . .	17
6.2	QxCppUnit Namespace Reference . . . . .	18
6.2.1	Detailed Description . . . . .	18
6.3	QxRunner Namespace Reference . . . . .	19
6.3.1	Detailed Description . . . . .	19
<b>7</b>	<b>Class Documentation</b>	<b>21</b>
7.1	CppUnitItem Class Reference . . . . .	21
7.1.1	Detailed Description . . . . .	21
7.1.2	Constructor & Destructor Documentation . . . . .	21
7.1.2.1	CppUnitItem . . . . .	21
7.1.2.2	~CppUnitItem . . . . .	21
7.1.3	Member Function Documentation . . . . .	21
7.1.3.1	run . . . . .	21
7.2	CppUnitModel Class Reference . . . . .	22
7.2.1	Detailed Description . . . . .	22
7.2.2	Constructor & Destructor Documentation . . . . .	22

7.2.2.1	CppUnitModel	22
7.2.2.2	~CppUnitModel	22
7.2.3	Member Function Documentation	22
7.2.3.1	about	22
7.2.3.2	addTest	22
7.2.3.3	name	22
7.3	TestRunner Class Reference	23
7.3.1	Detailed Description	23
7.3.2	Constructor & Destructor Documentation	23
7.3.2.1	TestRunner	23
7.3.2.2	~TestRunner	23
7.3.3	Member Function Documentation	23
7.3.3.1	addTest	23
7.3.3.2	addTests	23
7.3.3.3	run	23
<b>8</b>	<b>File Documentation</b>	<b>25</b>
8.1	cppunititem.cpp File Reference	25
8.1.1	Detailed Description	25
8.2	cppunititem.h File Reference	26
8.2.1	Detailed Description	26
8.3	cppunitmodel.cpp File Reference	27
8.3.1	Detailed Description	27
8.4	cppunitmodel.h File Reference	28
8.4.1	Detailed Description	28
8.5	mainpage.h File Reference	29
8.5.1	Detailed Description	29
8.6	qxcppunit_global.h File Reference	30
8.6.1	Detailed Description	30
8.6.2	Define Documentation	30
8.6.2.1	QXCPPUNIT_EXPORT	30
8.7	testrunner.cpp File Reference	31
8.7.1	Detailed Description	31
8.7.2	Function Documentation	31
8.7.2.1	initQxCppUnitResource	31
8.8	testrunner.h File Reference	32
8.8.1	Detailed Description	32

## Main Page

---

### 1.1 Introduction

The [QxCppUnit](#) library provides a GUI frontend for CppUnit test execution. It is based on the [QxRunner](#) and Qt libraries. It can be used to unit test any kind of C++ software, it's not limited to Qt applications.

This library is just a thin wrapper around the [QxRunner](#) library. Actually the idea to write a Qt test runner with a user-friendly GUI led to the development of the [QxRunner](#) library.

The [QxCppUnit](#) library mainly consists of the [CppUnitItem](#) class, which subclasses [QxRunner::RunnerItem](#), and the [CppUnitModel](#) class derived from [QxRunner::RunnerModel](#). The [TestRunner](#) class populates a [CppUnitModel](#) instance with [CppUnitItem](#) objects and uses a [QxRunner::Runner](#) instance to show the GUI on the screen.

To write a GUI test runner for CppUnit just the [TestRunner](#) class is needed as described in the [Examples](#) page.



**License**

---

QxCppUnit Library  
Copyright (C) 2006 systest.ch

This library is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 2.1 of the License, or (at your option) any later version.

This library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this library; if not, write to the Free Software Foundation, Inc., 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA

# **Implementation Aspects**

---

### 3.1 Exported Symbols

This applies for the Win32 world only. By default symbols of the library are 'hidden' when created as a DLL. Therefore symbols, in particular classes, that must be accessible from the outside of the library are exported with the `QXCPPUNIT_EXPORT` macro:

```
class QXCPPUNIT_EXPORT TestRunner
{
public: // Operations
    ...
}
```

Only the [TestRunner](#) class is exported right now, the other classes are for internal use only. This might be reconsidered in the future and more classes could become exported to be used somewhere outside the library.

### 3.2 Result Types

See the [CppUnitModel](#) class for the result types handled by the library.

## Examples

---

The demo program shipped with the [QxCppUnit](#) library can be used as the starting point for writing a GUI test runner for CppUnit. It has some CppUnit test fixtures with simple test cases which succeed or fail and uses a [TestRunner](#) instance in the main program to launch the GUI.

## 4.1 A CppUnit Test Example

This is one of the test fixtures of the demo program. Of course this is a very simple test and only used for demonstration purposes. It isn't intended to show how to write CppUnit tests.

```
1 // This class implements a set of CppUnit example test cases which
2 // succeed or fail. It isn't intended to show how to write CppUnit tests.
3
4 #include <cppunit/TestFixture.h>
5 #include <cppunit/extensions/HelperMacros.h>
6
7 class TestExamples2;
8
9 CPPUNIT_TEST_SUITE_REGISTRATION(TestExamples2);
10
11 class TestExamples2 : public CPPUNIT_NS::TestFixture
12 {
13     CPPUNIT_TEST_SUITE(TestExamples2);
14
15     CPPUNIT_TEST(testFail);
16     CPPUNIT_TEST(testAssertAssertionPass);
17     CPPUNIT_TEST(testAssertAssertionFail);
18     CPPUNIT_TEST(testUnhandledError);
19
20     CPPUNIT_TEST_SUITE_END();
21
22     void testFail()
23     {
24         CPPUNIT_FAIL("Intentionally failed");
25     }
26
27     void testAssertAssertionPass()
28     {
29         CPPUNIT_ASSERT_ASSERTION_PASS(CPPUNIT_ASSERT(1 == 1));
30     }
31
32     void testAssertAssertionFail()
33     {
34         CPPUNIT_ASSERT_ASSERTION_FAIL(CPPUNIT_ASSERT(1 == 2));
35     }
36
37     void testUnhandledError()
38     {
39         throw "Unhandled error";
40     }
41 };
```

## 4.2 The Demo Main Program

The main program is straightforward:

```
1 #include <QApplication>
2 #include <qxcppunit/testrunner.h>
3 #include <cppunit/extensions/TestFactoryRegistry.h>
4
5 int main(int argc, char *argv[])
6 {
7     QApplication app(argc, argv);
8
9     QxCppUnit::TestRunner runner;
10
11     runner.addTest(CPPUNIT_NS::TestFactoryRegistry::getRegistry().makeTest());
12     runner.run();
13 }
```

```
14     return 0;  
15 }
```



# Directory Documentation

---

## 5.1 qxrunner/include/ Directory Reference

### Directories

- directory [qxcppunit](#)

## 5.2 qxrunner/include/qxcppunit/ Directory Reference

### Files

- file [cppunititem.h](#)  
*Declares class CppUnitItem.*
- file [cppunitmodel.h](#)  
*Declares class CppUnitModel.*
- file [qxcppunit\\_global.h](#)  
*Declares global functions, macros, symbols and the like.*
- file [testrunner.h](#)  
*Declares class TestRunner.*

## 5.3 qxrunner/src/qxcppunit/ Directory Reference

### Files

- file [cppunititem.cpp](#)  
*Implements class CppUnitItem.*
- file [cppunitmodel.cpp](#)  
*Implements class CppUnitModel.*
- file [mainpage.h](#)  
*Contains documentation.*
- file [testrunner.cpp](#)  
*Implements class TestRunner.*

## 5.4 qxrunner/ Directory Reference

### Directories

- directory [include](#)
- directory [src](#)

## 5.5 qxrunner/src/ Directory Reference

### Directories

- directory [qxcppunit](#)



# Namespace Documentation

---

## 6.1 CPPUNIT\_NS Namespace Reference

The CppUnit library exposes its classes in the [CPPUNIT\\_NS](#) namespace.

### 6.1.1 Detailed Description

The CppUnit library exposes its classes in the [CPPUNIT\\_NS](#) namespace.

## 6.2 QxCppUnit Namespace Reference

Namespace for the [QxCppUnit](#) library entities.

### Classes

- class [CppUnitItem](#)  
*The [CppUnitItem](#) class executes a CppUnit test case.*
- class [CppUnitModel](#)  
*The [CppUnitModel](#) class maintains CppUnit data and executes CppUnit unit tests.*
- class [TestRunner](#)  
*The [TestRunner](#) class starts a [QxCppUnit](#) application.*

### 6.2.1 Detailed Description

Namespace for the [QxCppUnit](#) library entities.

## 6.3 QxRunner Namespace Reference

Namespace of the [QxRunner Library](#).

### 6.3.1 Detailed Description

Namespace of the [QxRunner Library](#).



# Class Documentation

---

## 7.1 CppUnitItem Class Reference

The [CppUnitItem](#) class executes a CppUnit test case.

```
#include <cppunititem.h>
```

### Public Member Functions

- [CppUnitItem](#) (const QList< QVariant > &data, RunnerItem \*parent=0, CPPUNIT\_NS::Test \*test=0)
- [~CppUnitItem](#) ()
- int [run](#) ()

### 7.1.1 Detailed Description

The [CppUnitItem](#) class executes a CppUnit test case.

The [CppUnitItem](#) class registers itself as a test listener and runs a CppUnit test. In case of failure the item columns contain the failure information, whereby the textual representation for a warning is set to 'failure' which complies with CppUnit speech. The class is designed to execute one individual test and not a test suite.

### 7.1.2 Constructor & Destructor Documentation

#### 7.1.2.1 CppUnitItem (const QList< QVariant > & *data*, RunnerItem \* *parent* = 0, CPPUNIT\_NS::Test \* *test* = 0)

Constructs a CppUnit item for one individual *test* with the test name in *data* and the given *parent*.

#### 7.1.2.2 ~CppUnitItem ()

Destroys this CppUnit item.

### 7.1.3 Member Function Documentation

#### 7.1.3.1 int run ()

Runs the test and returns the test result code which is of type QxRunner::RunnerResult.

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

- [cppunititem.h](#)
- [cppunititem.cpp](#)

## 7.2 CppUnitModel Class Reference

The [CppUnitModel](#) class maintains CppUnit data and executes CppUnit unit tests.

```
#include <cppunitmodel.h>
```

### Public Member Functions

- [CppUnitModel](#) (QObject \*parent=0)
- [~CppUnitModel](#) ()
- [QString name](#) () const
- [QString about](#) () const
- [void addTest](#) (CPPUNIT\_NS::Test \*test) const

### 7.2.1 Detailed Description

The [CppUnitModel](#) class maintains CppUnit data and executes CppUnit unit tests.

This class stores CppUnit test data using [CppUnitItem](#) objects. The CppUnit tests referenced by the [CppUnitItem](#) objects can be executed. Test results are prepared for representation in views.

The expected results from this model are:

- [QxRunner::RunSuccess](#) - CppUnit test case passed
- [QxRunner::RunWarning](#) - CppUnit assertion failed
- [QxRunner::RunError](#) - CppUnit reported an error

### 7.2.2 Constructor & Destructor Documentation

#### 7.2.2.1 CppUnitModel (QObject \* parent = 0)

Constructs a CppUnit model with the given *parent*. The root item is created and the set of expected results is defined.

#### 7.2.2.2 ~CppUnitModel ()

Destroys this CppUnit model.

### 7.2.3 Member Function Documentation

#### 7.2.3.1 QString about () const

Returns the CppUnit version used.

#### 7.2.3.2 void addTest (CPPUNIT\_NS::Test \* test) const

Appends the *test* to the list of CppUnit tests in the model. *test* can be one individual test or a test suite.

#### 7.2.3.3 QString name () const

Returns the model name "QxCppUnit".

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

- [cppunitmodel.h](#)
- [cppunitmodel.cpp](#)

## 7.3 TestRunner Class Reference

The [TestRunner](#) class starts a [QxCppUnit](#) application.

```
#include <testrunner.h>
```

### Public Member Functions

- [TestRunner](#) ()
- virtual [~TestRunner](#) ()
- void [addTest](#) (CPPUNIT\_NS::Test \*test) const
- void [addTests](#) (const CppUnitVector< CPPUNIT\_NS::Test \* > &tests) const
- void [run](#) ()

### 7.3.1 Detailed Description

The [TestRunner](#) class starts a [QxCppUnit](#) application.

This class creates a [CppUnitModel](#) and adds a [CppUnitItem](#) object for each given CppUnit test to the model. It sets an application icon and then shows the test runner main window on screen. This is the only class of the library that is used by clients to build a GUI test runner for CppUnit.

### 7.3.2 Constructor & Destructor Documentation

#### 7.3.2.1 TestRunner ()

Constructs a test runner. Creates the [CppUnitModel](#) instance.

#### 7.3.2.2 ~TestRunner () [virtual]

Destroys this test runner.

### 7.3.3 Member Function Documentation

#### 7.3.3.1 void addTest (CPPUNIT\_NS::Test \* test) const

Appends the *test* to the list of CppUnit tests. *test* can be one individual test or a test suite.

#### 7.3.3.2 void addTests (const CppUnitVector< CPPUNIT\_NS::Test \* > & tests) const

Appends the *tests* to the list of CppUnit tests. An entry in *tests* can be one individual test or a test suite.

#### 7.3.3.3 void run ()

Sets the application icon and shows the main window.

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

- [testrunner.h](#)
- [testrunner.cpp](#)



## File Documentation

---

### 8.1 cppunititem.cpp File Reference

Implements class CppUnitItem.

```
#include "cppunititem.h"
```

```
#include <cppunit/TestResult.h>
```

```
#include <cppunit/TestFailure.h>
```

```
#include <cppunit/Exception.h>
```

#### Namespaces

- namespace [QxCppUnit](#)  
*Namespace for the [QxCppUnit](#) library entities.*

#### 8.1.1 Detailed Description

Implements class CppUnitItem.

## 8.2 cppunititem.h File Reference

Declares class CppUnitItem.

```
#include <qxrunner/runneritem.h>
```

```
#include <cppunit/Test.h>
```

```
#include <cppunit/TestListener.h>
```

### Classes

- class [CppUnitItem](#)  
*The [CppUnitItem](#) class executes a CppUnit test case.*

### Namespaces

- namespace [QxCppUnit](#)  
*Namespace for the [QxCppUnit](#) library entities.*

### 8.2.1 Detailed Description

Declares class CppUnitItem.

## 8.3 cppunitmodel.cpp File Reference

Implements class CppUnitModel.

```
#include "cppunitmodel.h"
```

```
#include "cppunititem.h"
```

### Namespaces

- namespace [QxCppUnit](#)  
*Namespace for the [QxCppUnit](#) library entities.*

### 8.3.1 Detailed Description

Implements class CppUnitModel.

## 8.4 cppunitmodel.h File Reference

Declares class CppUnitModel.

```
#include <qxrunner/runnermodel.h>
```

```
#include <cppunit/Test.h>
```

### Classes

- class [CppUnitModel](#)

*The [CppUnitModel](#) class maintains CppUnit data and executes CppUnit unit tests.*

### Namespaces

- namespace [QxCppUnit](#)

*Namespace for the [QxCppUnit](#) library entities.*

### 8.4.1 Detailed Description

Declares class CppUnitModel.

## 8.5 mainpage.h File Reference

Contains documentation.

### Namespaces

- namespace [QxCppUnit](#)  
*Namespace for the [QxCppUnit](#) library entities.*
- namespace [QxRunner](#)  
*Namespace of the [QxRunner Library](#).*
- namespace [CPPUNIT\\_NS](#)  
*The CppUnit library exposes its classes in the [CPPUNIT\\_NS](#) namespace.*

### 8.5.1 Detailed Description

Contains documentation.

This file contains text and doxygen formatting commands for the [Main Page](#) and other parts of the API documentation.

## 8.6 qxcppunit\_global.h File Reference

Declares global functions, macros, symbols and the like.

```
#include <Qt/qglobal.h>
```

### Defines

- #define [QXCPPUNIT\\_EXPORT](#)

### 8.6.1 Detailed Description

Declares global functions, macros, symbols and the like.

### 8.6.2 Define Documentation

#### 8.6.2.1 #define QXCPPUNIT\_EXPORT

Macro to export symbols to DLL with VC++:

- QXCPPUNIT\_DLL\_BUILD must be defined when building the DLL.
- QXCPPUNIT\_DLL must be defined if linking against the DLL.
- If none of the above are defined then you are building or linking against the static library.

## 8.7 testrunner.cpp File Reference

Implements class TestRunner.

```
#include "testrunner.h"  
#include "cppunitmodel.h"  
#include <qxrunner/runner.h>  
#include <cppunit/TestSuite.h>  
#include <QIcon>
```

### Namespaces

- namespace [QxCppUnit](#)  
*Namespace for the [QxCppUnit](#) library entities.*

### Functions

- static void [initQxCppUnitResource](#) ()

#### 8.7.1 Detailed Description

Implements class TestRunner.

#### 8.7.2 Function Documentation

##### 8.7.2.1 [static void initQxCppUnitResource](#) () [static]

## 8.8 testrunner.h File Reference

Declares class TestRunner.

```
#include "qxcppunit_global.h"  
#include <cppunit/Portability.h>  
#include <cppunit/portability/CppUnitVector.h>
```

### Classes

- class [TestRunner](#)  
*The [TestRunner](#) class starts a [QxCppUnit](#) application.*

### Namespaces

- namespace [CPPUNIT\\_NS](#)  
*The [CppUnit](#) library exposes its classes in the [CPPUNIT\\_NS](#) namespace.*
- namespace [QxRunner](#)  
*Namespace of the [QxRunner](#) Library.*
- namespace [QxCppUnit](#)  
*Namespace for the [QxCppUnit](#) library entities.*

### 8.8.1 Detailed Description

Declares class TestRunner.

# Index

---

## - Symbols -

~CppUnitItem  
    QxCppUnit::CppUnitItem, 21  
~CppUnitModel  
    QxCppUnit::CppUnitModel, 22  
~TestRunner  
    QxCppUnit::TestRunner, 23

## - A -

about  
    QxCppUnit::CppUnitModel, 22  
addTest  
    QxCppUnit::CppUnitModel, 22  
    QxCppUnit::TestRunner, 23  
addTests  
    QxCppUnit::TestRunner, 23

## - C -

CPPUNIT\_NS, 17  
CppUnitItem  
    QxCppUnit::CppUnitItem, 21  
cppunititem.cpp, 25  
cppunititem.h, 26  
CppUnitModel  
    QxCppUnit::CppUnitModel, 22  
cppunitmodel.cpp, 27  
cppunitmodel.h, 28

## - I -

initQxCppUnitResource  
    testrunner.cpp, 31

## - M -

mainpage.h, 29

## - N -

name  
    QxCppUnit::CppUnitModel, 22

## - Q -

QxCppUnit, 18  
QxCppUnit::CppUnitItem, 21  
    ~CppUnitItem, 21  
    CppUnitItem, 21  
    run, 21  
QxCppUnit::CppUnitModel, 22  
    ~CppUnitModel, 22  
    about, 22  
    addTest, 22  
    CppUnitModel, 22  
    name, 22  
QxCppUnit::TestRunner, 23  
    ~TestRunner, 23  
    addTest, 23  
    addTests, 23  
    run, 23  
    TestRunner, 23  
QXCPPUNIT\_EXPORT  
    qxcppunit\_global.h, 30  
qxcppunit\_global.h, 30

QXCPPUNIT\_EXPORT, 30  
QxRunner, 19  
qxrunner/ Directory Reference, 14  
qxrunner/include/ Directory Reference, 11  
qxrunner/include/qxcppunit/ Directory Reference, 12  
qxrunner/src/ Directory Reference, 15  
qxrunner/src/qxcppunit/ Directory Reference, 13

## - R -

run  
    QxCppUnit::CppUnitItem, 21  
    QxCppUnit::TestRunner, 23

## - T -

TestRunner  
    QxCppUnit::TestRunner, 23  
testrunner.cpp, 31  
    initQxCppUnitResource, 31  
testrunner.h, 32