

KText Editor

Generated by Doxygen 1.9.6

1 General Overview	1
1.1 Features	1
1.2 Order i should implement things	1
1.3 Structure	2
2 Namespace Index	3
2.1 Namespace List	3
3 File Index	5
3.1 File List	5
4 Namespace Documentation	7
4.1 Command Namespace Reference	7
4.2 hey Namespace Reference	7
5 File Documentation	9
5.1 src/Document.cpp File Reference	9
5.2 src/Editor.cpp File Reference	9
5.3 src/kamil.cpp File Reference	9
5.3.1 Function Documentation	10
5.3.1.1 main()	10
5.4 src/Keyboard.cpp File Reference	10
5.5 src/MyRect.cpp File Reference	10
5.6 src/Notes.md File Reference	10
5.7 src/test/another.cpp File Reference	10
5.8 src/test/build/CMakeFiles/3.26.2/CompilerIdC/CMakeCCompilerId.c File Reference	11
5.8.1 Macro Definition Documentation	11
5.8.1.1 __has_include	11
5.8.1.2 ARCHITECTURE_ID	11
5.8.1.3 C_VERSION	12
5.8.1.4 COMPILER_ID	12
5.8.1.5 DEC	12
5.8.1.6 HEX	12
5.8.1.7 PLATFORM_ID	12
5.8.1.8 STRINGIFY	13
5.8.1.9 STRINGIFY_HELPER	13
5.8.2 Function Documentation	13
5.8.2.1 main()	13
5.8.3 Variable Documentation	13
5.8.3.1 info_arch	13
5.8.3.2 info_compiler	13
5.8.3.3 info_language_extensions_default	13
5.8.3.4 info_language_standard_default	14
5.8.3.5 info_platform	14

5.9 src/test/build/CMakeFiles/3.26.2/CompilerIdCXX/CMakeCXXCompilerId.cpp File Reference	14
5.9.1 Macro Definition Documentation	14
5.9.1.1 __has_include	15
5.9.1.2 ARCHITECTURE_ID	15
5.9.1.3 COMPILER_ID	15
5.9.1.4 CXX_STD	15
5.9.1.5 DEC	15
5.9.1.6 HEX	15
5.9.1.7 PLATFORM_ID	16
5.9.1.8 STRINGIFY	16
5.9.1.9 STRINGIFY_HELPER	16
5.9.2 Function Documentation	16
5.9.2.1 main()	16
5.9.3 Variable Documentation	16
5.9.3.1 info_arch	16
5.9.3.2 info_compiler	16
5.9.3.3 info_language_extensions_default	17
5.9.3.4 info_language_standard_default	17
5.9.3.5 info_platform	17
5.10 src/test/build/CMakeFiles/test.dir/main.cpp.o.d File Reference	17
5.11 src/test/hey.py File Reference	17
5.12 src/test/main.cpp File Reference	18
5.12.1 Function Documentation	18
5.12.1.1 deleteChar()	18
5.12.1.2 deleteOtherChar()	18
5.12.1.3 main()	18
5.13 src/test/test.cpp File Reference	19
5.13.1 Function Documentation	19
5.13.1.1 main()	19
5.14 src/TextBox.cpp File Reference	19
5.15 src/Utils/Stack.cpp File Reference	19
5.16 src/Utils/tet.cpp File Reference	19
5.16.1 Function Documentation	20
5.16.1.1 main()	20
Index	21

Chapter 1

General Overview

- I want a text editor similar to vim

1.1 Features

- modal
 - **Command** mode, Insert mode
- show lines at side of window
- Input and delete text
 - use backspace and hjkl to move through text
- working cursor
 - change colour
- undo using commands
- Save file
- move using mouse

1.2 Order i should implement things

- Make window
- section window in two for lines and text box
- add cursor to text box side

1.3 Structure

Kamil.cpp - Main window (text window is used) -Window (has the text box window and the line number) -inputs (handles events like keyboard clicks and mouse events, state changes happen here) -textdoc (handles the file like saving it)

window

- Main Window Editor
- Where text is handled (same size as window)
- Rectangle on left for margin
- font
- line info
- drawn on main window Input
- mouse and keyboard window TextDocument

Chapter 2

Namespace Index

2.1 Namespace List

Here is a list of all namespaces with brief descriptions:

Command	7
hey	7

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

src/ Document.cpp	9
src/ Editor.cpp	9
src/ kamil.cpp	9
src/ Keyboard.cpp	10
src/ MyRect.cpp	10
src/ TextBox.cpp	19
src/test/ another.cpp	10
src/test/ hey.py	17
src/test/ main.cpp	18
src/test/ test.cpp	19
src/test/build/CMakeFiles/3.26.2/CompilerIdC/ CMakeCCompilerId.c	11
src/test/build/CMakeFiles/3.26.2/CompilerIdCXX/ CMakeCXXCompilerId.cpp	14
src/test/build/CMakeFiles/test.dir/ main.cpp.o.d	17
src/Utils/ Stack.cpp	19
src/Utils/ tet.cpp	19

Chapter 4

Namespace Documentation

4.1 Command Namespace Reference

4.2 hey Namespace Reference

Chapter 5

File Documentation

5.1 src/Document.cpp File Reference

```
#include <Kamil/Document.h>
#include <cstdio>
#include <cstdlib>
#include <fmt/core.h>
#include <iostream>
#include <sstream>
```

Include dependency graph for Document.cpp:

5.2 src/Editor.cpp File Reference

```
#include <Kamil/Editor.h>
#include <SFML/Config.hpp>
#include <SFML/Graphics/Color.hpp>
#include <SFML/Graphics/RectangleShape.hpp>
#include <SFML/Graphics/RenderStates.hpp>
#include <SFML/Graphics/RenderWindow.hpp>
#include <SFML/System/Vector2.hpp>
#include <fmt/core.h>
```

Include dependency graph for Editor.cpp:

5.3 src/kamil.cpp File Reference

```
#include "Kamil/TextBox.h"
#include <SFML/Window/Event.hpp>
#include <SFML/Window/Keyboard.hpp>
#include <iostream>
#include <Kamil/Editor.h>
#include <fmt/core.h>
#include <Kamil/Document.h>
```

Include dependency graph for kamil.cpp:

Functions

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

5.3.1 Function Documentation

5.3.1.1 `main()`

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

5.4 `src/Keyboard.cpp` File Reference

```
#include <Kamil/Keyboard.h>  
#include <Kamil/Editor.h>  
#include <SFML/System/Vector2.hpp>  
#include <SFML/Window/Event.hpp>  
#include <SFML/Window/Keyboard.hpp>
```

Include dependency graph for `Keyboard.cpp`:

5.5 `src/MyRect.cpp` File Reference

```
#include <Kamil/MyRect.h>  
#include <SFML/Graphics/Color.hpp>  
#include <SFML/Graphics/RectangleShape.hpp>  
#include <SFML/Graphics/RenderStates.hpp>  
#include <SFML/System/Vector2.hpp>
```

Include dependency graph for `MyRect.cpp`:

5.6 `src/Notes.md` File Reference

5.7 `src/test/another.cpp` File Reference

```
#include <iostream>  
#include <cstdlib>  
#include <fstream>
```

Include dependency graph for `another.cpp`:

5.8 src/test/build/CMakeFiles/3.26.2/CompilerIdC/CMakeCCompilerId.c File Reference

Macros

- `#define __has_include(x) 0`
- `#define COMPILER_ID ""`
- `#define STRINGIFY_HELPER(X) #X`
- `#define STRINGIFY(X) STRINGIFY_HELPER(X)`
- `#define PLATFORM_ID`
- `#define ARCHITECTURE_ID`
- `#define DEC(n)`
- `#define HEX(n)`
- `#define C_VERSION`

Functions

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

Variables

- `char const * info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"`
- `char const * info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"`
- `char const * info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"`
- `const char * info_language_standard_default`
- `const char * info_language_extensions_default`

5.8.1 Macro Definition Documentation

5.8.1.1 __has_include

```
#define __has_include(  
    x ) 0
```

5.8.1.2 ARCHITECTURE_ID

```
#define ARCHITECTURE_ID
```

5.8.1.3 C_VERSION

```
#define C_VERSION
```

5.8.1.4 COMPILER_ID

```
#define COMPILER_ID ""
```

5.8.1.5 DEC

```
#define DEC(  
    n )
```

Value:

```
('0' + ((n) / 10000000)%10), \
('0' + ((n) / 1000000)%10), \
('0' + ((n) / 100000)%10), \
('0' + ((n) / 10000)%10), \
('0' + ((n) / 1000)%10), \
('0' + ((n) / 100)%10), \
('0' + ((n) / 10)%10), \
('0' + ((n) % 10))
```

5.8.1.6 HEX

```
#define HEX(  
    n )
```

Value:

```
('0' + ((n)>>28 & 0xF)), \
('0' + ((n)>>24 & 0xF)), \
('0' + ((n)>>20 & 0xF)), \
('0' + ((n)>>16 & 0xF)), \
('0' + ((n)>>12 & 0xF)), \
('0' + ((n)>>8 & 0xF)), \
('0' + ((n)>>4 & 0xF)), \
('0' + ((n) & 0xF))
```

5.8.1.7 PLATFORM_ID

```
#define PLATFORM_ID
```


5.8.1.8 STRINGIFY

```
#define STRINGIFY(  
    X ) STRINGIFY_HELPER(X)
```

5.8.1.9 STRINGIFY_HELPER

```
#define STRINGIFY_HELPER(  
    X ) #X
```

5.8.2 Function Documentation

5.8.2.1 main()

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

5.8.3 Variable Documentation

5.8.3.1 info_arch

```
char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
```

5.8.3.2 info_compiler

```
char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
```

5.8.3.3 info_language_extensions_default

```
const char* info_language_extensions_default
```

Initial value:

```
= "INFO" ":" "extensions_default["
```

```
    "OFF"  
    "]"
```

5.8.3.4 info_language_standard_default

```
const char* info_language_standard_default
```

Initial value:

```
= "INFO" ":" "standard_default[" C_VERSION "]"
```

5.8.3.5 info_platform

```
char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
```

5.9 src/test/build/CMakeFiles/3.26.2/CompilerIdCXX/CMakeCXXCompilerId.cpp File Reference

Macros

- #define `__has_include(x)` 0
- #define `COMPILER_ID` ""
- #define `STRINGIFY_HELPER(X)` #X
- #define `STRINGIFY(X)` `STRINGIFY_HELPER(X)`
- #define `PLATFORM_ID`
- #define `ARCHITECTURE_ID`
- #define `DEC(n)`
- #define `HEX(n)`
- #define `CXX_STD` `__cplusplus`

Functions

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

Variables

- char const * `info_compiler` = "INFO" ":" "compiler[" `COMPILER_ID` "]"
- char const * `info_platform` = "INFO" ":" "platform[" `PLATFORM_ID` "]"
- char const * `info_arch` = "INFO" ":" "arch[" `ARCHITECTURE_ID` "]"
- const char * `info_language_standard_default`
- const char * `info_language_extensions_default`

5.9.1 Macro Definition Documentation

5.9.1.1 __has_include

```
#define __has_include(
    x ) 0
```

5.9.1.2 ARCHITECTURE_ID

```
#define ARCHITECTURE_ID
```

5.9.1.3 COMPILER_ID

```
#define COMPILER_ID ""
```

5.9.1.4 CXX_STD

```
#define CXX_STD __cplusplus
```

5.9.1.5 DEC

```
#define DEC(
    n )
```

Value:

```
('0' + ((n) / 10000000) % 10), \
('0' + ((n) / 1000000) % 10), \
('0' + ((n) / 100000) % 10), \
('0' + ((n) / 10000) % 10), \
('0' + ((n) / 1000) % 10), \
('0' + ((n) / 100) % 10), \
('0' + ((n) / 10) % 10), \
('0' + ((n) % 10))
```

5.9.1.6 HEX

```
#define HEX(
    n )
```

Value:

```
('0' + ((n) >> 28 & 0xF)), \
('0' + ((n) >> 24 & 0xF)), \
('0' + ((n) >> 20 & 0xF)), \
('0' + ((n) >> 16 & 0xF)), \
('0' + ((n) >> 12 & 0xF)), \
('0' + ((n) >> 8 & 0xF)), \
('0' + ((n) >> 4 & 0xF)), \
('0' + ((n) & 0xF))
```

5.9.1.7 PLATFORM_ID

```
#define PLATFORM_ID
```

5.9.1.8 STRINGIFY

```
#define STRINGIFY(  
    X ) STRINGIFY_HELPER(X)
```

5.9.1.9 STRINGIFY_HELPER

```
#define STRINGIFY_HELPER(  
    X ) #X
```

5.9.2 Function Documentation

5.9.2.1 main()

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

5.9.3 Variable Documentation

5.9.3.1 info_arch

```
char const* info_arch = "INFO" ":" "arch[" ARCHITECTURE_ID "]"
```

5.9.3.2 info_compiler

```
char const* info_compiler = "INFO" ":" "compiler[" COMPILER_ID "]"
```

5.9.3.3 info_language_extensions_default

```
const char* info_language_extensions_default
```

Initial value:

```
= "INFO" ":" "extensions_default["
```

```
    "OFF"  
"]"
```

5.9.3.4 info_language_standard_default

```
const char* info_language_standard_default
```

Initial value:

```
= "INFO" ":" "standard_default["
```

```
    "98"  
"]"
```

5.9.3.5 info_platform

```
char const* info_platform = "INFO" ":" "platform[" PLATFORM_ID "]"
```

5.10 src/test/build/CMakeFiles/test.dir/main.cpp.o.d File Reference

5.11 src/test/hey.py File Reference

Namespaces

- namespace [hey](#)

5.12 src/test/main.cpp File Reference

```
#include <SFML/Graphics/Color.hpp>
#include <cstdlib>
#include <iostream>
#include <sstream>
#include <string>
#include <unistd.h>
#include <SFML/Graphics.hpp>
Include dependency graph for main.cpp:
```

Functions

- void [deleteChar](#) (int char_typed, std::ostringstream &inputStringBuffer, std::ostringstream *deleteString↵ Buffer)
- void [deleteOtherChar](#) (int char_typed, sf::Text *text, std::string &inputStringBuffer, std::string *deleteString↵ Buffer)
- int [main](#) ()

5.12.1 Function Documentation

5.12.1.1 deleteChar()

```
void deleteChar (
    int char_typed,
    std::ostringstream & inputStringBuffer,
    std::ostringstream * deleteStringBuffer )
```

5.12.1.2 deleteOtherChar()

```
void deleteOtherChar (
    int char_typed,
    sf::Text * text,
    std::string & inputStringBuffer,
    std::string * deleteStringBuffer )
```

Here is the caller graph for this function:

5.12.1.3 main()

```
int main ( )
```

Here is the call graph for this function:

5.13 src/test/test.cpp File Reference

```
#include <iostream>
#include <string>
#include <sstream>
#include <fmt/core.h>
Include dependency graph for test.cpp:
```

Functions

- int [main](#) ()

5.13.1 Function Documentation

5.13.1.1 main()

```
int main ( )
```

5.14 src/TextBox.cpp File Reference

```
#include "Kamil/MyRect.h"
#include <Kamil/TextBox.h>
#include <SFML/Graphics/RectangleShape.hpp>
#include <SFML/Graphics/RenderStates.hpp>
#include <SFML/System/Vector2.hpp>
#include <SFML/Window/Keyboard.hpp>
Include dependency graph for TextBox.cpp:
```

5.15 src/Utils/Stack.cpp File Reference

```
#include <Kamil/Utils/Stack.h>
Include dependency graph for Stack.cpp:
```

Namespaces

- namespace [Command](#)

5.16 src/Utils/tet.cpp File Reference

```
#include "../..//include/Kamil/Utils/Stack.h"
Include dependency graph for tet.cpp:
```

Functions

- int `main` ()

5.16.1 Function Documentation

5.16.1.1 `main()`

```
int main ( )
```


Index

- [__has_include](#)
 - [CMakeCCompilerId.c, 11](#)
 - [CMakeCXXCompilerId.cpp, 14](#)
- [ARCHITECTURE_ID](#)
 - [CMakeCCompilerId.c, 11](#)
 - [CMakeCXXCompilerId.cpp, 15](#)
- [C_VERSION](#)
 - [CMakeCCompilerId.c, 11](#)
- [CMakeCCompilerId.c](#)
 - [__has_include, 11](#)
 - [ARCHITECTURE_ID, 11](#)
 - [C_VERSION, 11](#)
 - [COMPILER_ID, 12](#)
 - [DEC, 12](#)
 - [HEX, 12](#)
 - [info_arch, 13](#)
 - [info_compiler, 13](#)
 - [info_language_extensions_default, 13](#)
 - [info_language_standard_default, 13](#)
 - [info_platform, 14](#)
 - [main, 13](#)
 - [PLATFORM_ID, 12](#)
 - [STRINGIFY, 12](#)
 - [STRINGIFY_HELPER, 13](#)
- [CMakeCXXCompilerId.cpp](#)
 - [__has_include, 14](#)
 - [ARCHITECTURE_ID, 15](#)
 - [COMPILER_ID, 15](#)
 - [CXX_STD, 15](#)
 - [DEC, 15](#)
 - [HEX, 15](#)
 - [info_arch, 16](#)
 - [info_compiler, 16](#)
 - [info_language_extensions_default, 16](#)
 - [info_language_standard_default, 17](#)
 - [info_platform, 17](#)
 - [main, 16](#)
 - [PLATFORM_ID, 15](#)
 - [STRINGIFY, 16](#)
 - [STRINGIFY_HELPER, 16](#)
- [Command, 7](#)
- [COMPILER_ID](#)
 - [CMakeCCompilerId.c, 12](#)
 - [CMakeCXXCompilerId.cpp, 15](#)
- [CXX_STD](#)
 - [CMakeCXXCompilerId.cpp, 15](#)
- [DEC](#)
 - [CMakeCCompilerId.c, 12](#)
 - [CMakeCXXCompilerId.cpp, 15](#)
- [deleteChar](#)
 - [main.cpp, 18](#)
- [deleteOtherChar](#)
 - [main.cpp, 18](#)
- [HEX](#)
 - [CMakeCCompilerId.c, 12](#)
 - [CMakeCXXCompilerId.cpp, 15](#)
- [hey, 7](#)
- [info_arch](#)
 - [CMakeCCompilerId.c, 13](#)
 - [CMakeCXXCompilerId.cpp, 16](#)
- [info_compiler](#)
 - [CMakeCCompilerId.c, 13](#)
 - [CMakeCXXCompilerId.cpp, 16](#)
- [info_language_extensions_default](#)
 - [CMakeCCompilerId.c, 13](#)
 - [CMakeCXXCompilerId.cpp, 16](#)
- [info_language_standard_default](#)
 - [CMakeCCompilerId.c, 13](#)
 - [CMakeCXXCompilerId.cpp, 17](#)
- [info_platform](#)
 - [CMakeCCompilerId.c, 14](#)
 - [CMakeCXXCompilerId.cpp, 17](#)
- [kamil.cpp](#)
 - [main, 10](#)
- [main](#)
 - [CMakeCCompilerId.c, 13](#)
 - [CMakeCXXCompilerId.cpp, 16](#)
 - [kamil.cpp, 10](#)
 - [main.cpp, 18](#)
 - [test.cpp, 19](#)
 - [tet.cpp, 20](#)
- [main.cpp](#)
 - [deleteChar, 18](#)
 - [deleteOtherChar, 18](#)
 - [main, 18](#)
- [PLATFORM_ID](#)
 - [CMakeCCompilerId.c, 12](#)
 - [CMakeCXXCompilerId.cpp, 15](#)
- [src/Document.cpp, 9](#)
- [src/Editor.cpp, 9](#)
- [src/kamil.cpp, 9](#)
- [src/Keyboard.cpp, 10](#)

- src/MyRect.cpp, [10](#)
- src/Notes.md, [10](#)
- src/test/another.cpp, [10](#)
- src/test/build/CMakeFiles/3.26.2/CompilerIdC/CMakeCCompilerId.c,
[11](#)
- src/test/build/CMakeFiles/3.26.2/CompilerIdCXX/CMakeCXXCompilerId.cpp,
[14](#)
- src/test/build/CMakeFiles/test.dir/main.cpp.o.d, [17](#)
- src/test/hey.py, [17](#)
- src/test/main.cpp, [18](#)
- src/test/test.cpp, [19](#)
- src/TextBox.cpp, [19](#)
- src/Utils/Stack.cpp, [19](#)
- src/Utils/tet.cpp, [19](#)
- STRINGIFY
 - CMakeCCompilerId.c, [12](#)
 - CMakeCXXCompilerId.cpp, [16](#)
- STRINGIFY_HELPER
 - CMakeCCompilerId.c, [13](#)
 - CMakeCXXCompilerId.cpp, [16](#)
- test.cpp
 - main, [19](#)
- tet.cpp
 - main, [20](#)