

Kononov

Создано системой Doxygen 1.9.1

1 Иерархический список классов	1
1.1 Иерархия классов	1
2 Алфавитный указатель классов	3
2.1 Классы	3
3 Список файлов	5
3.1 Файлы	5
4 Классы	7
4.1 Класс Client	7
4.1.1 Подробное описание	7
4.1.2 Методы	8
4.1.2.1 Server()	8
4.2 Класс client_error	8
4.2.1 Подробное описание	9
5 Файлы	11
5.1 Файл Client.cpp	11
5.1.1 Подробное описание	11
5.2 Файл Client.h	11
5.2.1 Подробное описание	12
5.3 Файл main.cpp	13
5.3.1 Подробное описание	13
5.4 Файл md5.cpp	14
5.4.1 Подробное описание	14
5.4.2 Функции	14
5.4.2.1 MD5_hash()	14
5.5 Файл md5.h	15
5.5.1 Подробное описание	16
5.5.2 Функции	16
5.5.2.1 MD5_hash()	16
Предметный указатель	17

Глава 1

Иерархический список классов

1.1 Иерархия классов

Иерархия классов.

Client	7
invalid_argument	
client_error	8

Глава 2

Алфавитный указатель классов

2.1 Классы

Классы с их кратким описанием.

Client	Класс для подключения к серверу	7
client_error	Класс обработки ошибок client_error	8

Глава 3

Список файлов

3.1 Файлы

Полный список документированных файлов.

Client.cpp	Файл взаимодействие с сервером	11
Client.h	Заголовочный файл для модуля Client	11
main.cpp	Главный модуль программы для получения параметров от пользователя	13
md5.cpp	Файл для хэширования md5	14
md5.h	Заголовочный файл для модуля md5	15

Глава 4

Классы

4.1 Класс Client

Класс для подключения к серверу

```
#include <Client.h>
```

Открытые члены

- int [Server](#) (string str1, string str2)
Взаимодействие с сервером

Открытые атрибуты

- int port
- uint32_t sum
- string msg
- string vector_file
- string resultat_file
- string autf_file
- string username
- string pswd
- string line

4.1.1 Подробное описание

Класс для подключения к серверу

Аргументы

port	порт
sum	сумма
msg	отправляемый хэш
vector_file	имя файла для считывания векторов
resultat_file	имя файла для записи суммы векторов
autf_file	имя файла, в котором хранятся логины и пароли клиентов
username	логин клиента
line	строка из файла

4.1.2 Методы

4.1.2.1 Server()

```
int Client::Server (
    string str1,
    string str2 )
```

Взаимодействие с сервером

Аргументы

str1	адрес сервера
str2	порт сервера

Исключения

client_error	класс вызываемый при возникновении ошибки
------------------------------	---

Объявления и описания членов классов находятся в файлах:

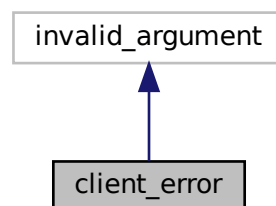
- [Client.h](#)
- [Client.cpp](#)

4.2 Класс client_error

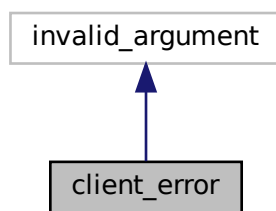
Класс обработки ошибок [client_error](#).

```
#include <Client.h>
```

Граф наследования:client_error:



Граф связей класса `client_error`:



Открытые члены

- `client_error (const string &what_arg)`
- `client_error (const char *what_arg)`

4.2.1 Подробное описание

Класс обработки ошибок [client_error](#).

Класс выводящий сообщения об ошибках

Объявления и описания членов класса находятся в файле:

- [Client.h](#)

Глава 5

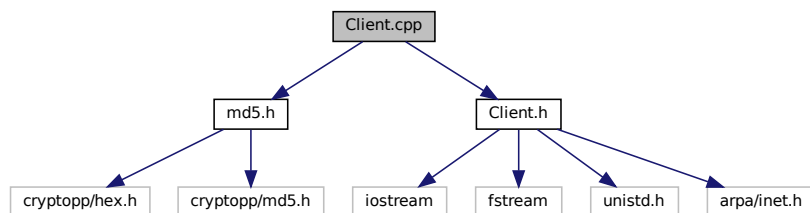
Файлы

5.1 Файл Client.cpp

Файл взаимодействие с сервером

```
#include "md5.h"  
#include "Client.h"
```

Граф включаемых заголовочных файлов для Client.cpp:



5.1.1 Подробное описание

Файл взаимодействие с сервером

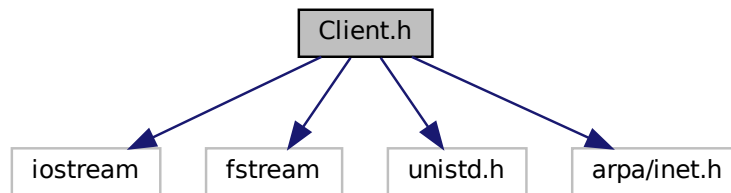
5.2 Файл Client.h

Заголовочный файл для модуля [Client](#).

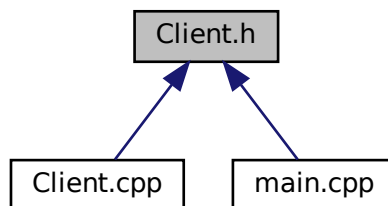
```
#include <iostream>  
#include <fstream>  
#include <unistd.h>
```

```
#include <arpa/inet.h>
```

Граф включаемых заголовочных файлов для Client.h:



Граф файлов, в которые включается этот файл:



Классы

- class [Client](#)
Класс для подключения к серверу
- class [client_error](#)
Класс обработки ошибок [client_error](#).

5.2.1 Подробное описание

Заголовочный файл для модуля [Client](#).

Автор

Kononov

Версия

1.0

Дата

4.05.2023

Авторство

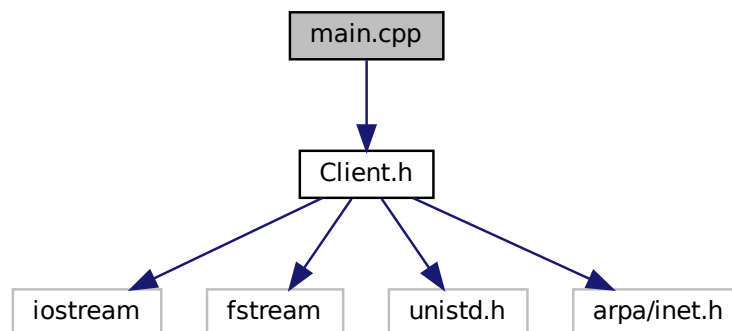
ИБСТ ПГУ

5.3 Файл main.cpp

Главный модуль программы для получения параметров от пользователя

```
#include "Client.h"
```

Граф включаемых заголовочных файлов для main.cpp:



Функции

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

5.3.1 Подробное описание

Главный модуль программы для получения параметров от пользователя

Аргументы

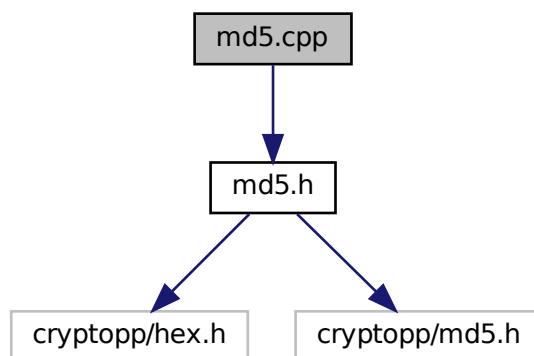
opt	переменная для работы с параметрами командной строки
optarg	переменная для получения параметров командной строки

5.4 Файл md5.cpp

Файл для хэширования md5.

```
#include "md5.h"
```

Граф включаемых заголовочных файлов для md5.cpp:



Функции

- string `MD5_hash` (string hsh)
Получение хэша по алгоритму md5.

5.4.1 Подробное описание

Файл для хэширования md5.

5.4.2 Функции

5.4.2.1 MD5_hash()

```
string MD5_hash (  
    string hsh )
```

Получение хэша по алгоритму md5.

Аргументы

hsh	сообщения получаемое для хэширования
-----	--------------------------------------

Возвращает

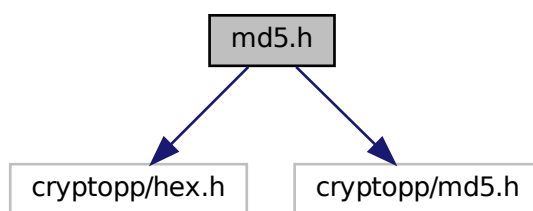
Полученный хэш

5.5 Файл md5.h

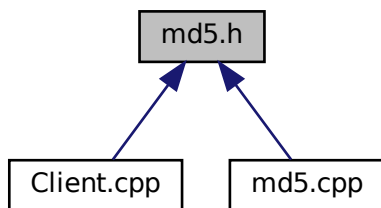
Заголовочный файл для модуля md5.

```
#include <cryptopp/hex.h>
#include <cryptopp/md5.h>
```

Граф включаемых заголовочных файлов для md5.h:



Граф файлов, в которые включается этот файл:



Макросы

- `#define CRYPTOPP_ENABLE_NAMESPACE_WEAK 1`

Функции

- `string MD5_hash (string hsh)`
Получение хэша по алгоритму md5.

5.5.1 Подробное описание

Заголовочный файл для модуля md5.

Автор

Kononov

Версия

1.0

Дата

04.05.2023

Авторство

ИБСТ ПГУ

5.5.2 Функции

5.5.2.1 MD5_hash()

```
string MD5_hash (  
    string hsh )
```

Получение хэша по алгоритму md5.

Аргументы

hsh	сообщения получаемое для хэширования
-----	--------------------------------------

Возвращает

Полученный хэш

Предметный указатель

- Client, [7](#)
 - Server, [8](#)
- Client.cpp, [11](#)
- Client.h, [11](#)
- client_error, [8](#)
- main.cpp, [13](#)
- md5.cpp, [14](#)
 - MD5_hash, [14](#)
- md5.h, [15](#)
 - MD5_hash, [16](#)
- MD5_hash
 - md5.cpp, [14](#)
 - md5.h, [16](#)
- Server
 - Client, [8](#)