

Сервер

Создано системой Doxygen 1.9.4

1	Алфавитный указатель классов	1
1.1	Классы	1
2	Список файлов	3
2.1	Файлы	3
3	Классы	5
3.1	Класс Auth	5
3.1.1	Подробное описание	5
3.2	Класс Calculator	5
3.2.1	Подробное описание	6
3.3	Класс CommandLineInterface	6
3.3.1	Подробное описание	6
3.3.2	Методы	6
3.3.2.1	printHelp()	6
3.4	Класс Logger	7
3.4.1	Подробное описание	7
3.5	Класс Server	7
4	Файлы	9
4.1	auth.h	9
4.2	calc.h	9
4.3	interface.h	9
4.4	logger.h	10
4.5	Файл server.cpp	10
4.5.1	Подробное описание	10
4.6	server.h	11
	Предметный указатель	13

Глава 1

Алфавитный указатель классов

1.1 Классы

Классы с их кратким описанием.

Auth	Класс Auth отвечает за аутентификацию на сервере	5
Calculator	Класс Calculator отвечает за вычисления сервера	5
CommandLineInterface	Класс CommandLineInterface отвечает за интерфейс сервера	6
Logger	Класс Logger отвечает за log.txt	7
Server	7

Глава 2

Список файлов

2.1 Файлы

Полный список документированных файлов.

auth.h	??
calc.h	??
interface.h	??
logger.h	??
server.cpp	
Главный файл программы	10
server.h	??

Глава 3

Классы

3.1 Класс Auth

Класс `Auth` отвечает за аутентификацию на сервере

```
#include <auth.h>
```

Открытые члены

- `bool login (const std::string &username, const std::string &password)`

Закрытые статические данные

- `static const std::string validUsername = "user"`
- `static const std::string validPassword = "P@ssW0rd"`

3.1.1 Подробное описание

Класс `Auth` отвечает за аутентификацию на сервере

Объявления и описания членов классов находятся в файлах:

- `auth.h`
- `auth.cpp`

3.2 Класс Calculator

Класс `Calculator` отвечает за вычисления сервера

```
#include <calc.h>
```

Открытые статические члены

- static double calculate_average (const std::vector< double > &vec)
 - Считает среднее арифметическое вектора

3.2.1 Подробное описание

Класс [Calculator](#) отвечает за вычисления сервера

Объявления и описания членов классов находятся в файлах:

- calc.h
- calc.cpp

3.3 Класс CommandLineInterface

Класс [CommandLineInterface](#) отвечает за интерфейс сервера

```
#include <interface.h>
```

Открытые члены

- void [printHelp](#) () const
 - Выводит справку
- void handleArguments (int argc, char *argv[])
 - Отвечает за интерфейс сервера

3.3.1 Подробное описание

Класс [CommandLineInterface](#) отвечает за интерфейс сервера

Аргументы

printHelp	
-----------	--

3.3.2 Методы

3.3.2.1 printHelp()

```
void CommandLineInterface::printHelp ( ) const
```

- Выводит справку

Аргументы

-h,-v,-p	
----------	--

Объявления и описания членов классов находятся в файлах:

- interface.h
- interface.cpp

3.4 Класс Logger

Класс `Logger` отвечает за log.txt.

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

Открытые члены

- `Logger (const std::string &logFile)`
- `void log (const std::string &message)`
 - Отвечает за файл log.txt, записывает в него ошибки программы

Закрытые данные

- `std::string logFile`

3.4.1 Подробное описание

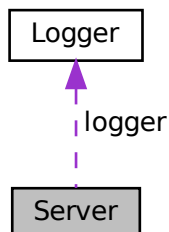
Класс `Logger` отвечает за log.txt.

Объявления и описания членов классов находятся в файлах:

- logger.h
- logger.cpp

3.5 Класс Server

Граф связей класса Server:



Открытые члены

- `Server (const std::string &configFile, int port, Logger &logger)`
- `void start ()`

Закрытые данные

- `std::string configFile`
- `int port`
- `Logger & logger`

Объявления и описания членов класса находятся в файле:

- `server.h`

Глава 4

Файлы

4.1 auth.h

```
1 #ifndef AUTH_H
2 #define AUTH_H
3
4 #include <string>
5
6 class Auth {
7 public:
8     // Конструктор
9     Auth();
10
11     // Функция для авторизации
12     bool login(const std::string& username, const std::string& password);
13
14 private:
15     // Статические данные для проверки логина и пароля
16     static const std::string validUsername;
17     static const std::string validPassword;
18 };
19
20 #endif // AUTH_H
```

4.2 calc.h

```
1 #ifndef CALC_H
2 #define CALC_H
3
4 #include <vector>
5
6 class Calculator {
7 public:
8     static double calculate_average(const std::vector<double>& vec);
9 };
10
11 #endif // CALC_H
```

4.3 interface.h

```
1 #ifndef INTERFACE_H
2 #define INTERFACE_H
3
4 #include <string>
5
6 class CommandLineInterface {
7 public:
8     void printHelp() const; // Метод для вывода справки
9     void handleArguments(int argc, char* argv[]); // Метод для обработки аргументов командной строки
10 };
11
12 #endif // INTERFACE_H
```

4.4 logger.h

```

1 #ifndef LOGGER_H
2 #define LOGGER_H
3
4 #include <string>
5 class Logger {
6 public:
7     Logger(const std::string& logFile);
8     void log(const std::string& message);
9
10 private:
11     std::string logFile;
12 };
13 #endif // LOGGER_H

```

4.5 Файл server.cpp

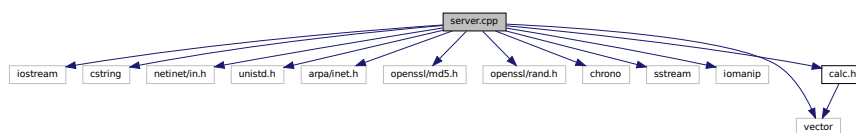
Главный файл программы.

```

#include <iostream>
#include <cstring>
#include <netinet/in.h>
#include <unistd.h>
#include <arpa/inet.h>
#include <openssl/md5.h>
#include <openssl/rand.h>
#include <chrono>
#include <sstream>
#include <iomanip>
#include <vector>
#include "calc.h"

```

Граф включаемых заголовочных файлов для server.cpp:



Функции

- std::string generate_salt ()
- std::string hash_password (const std::string &password, const std::string &salt)
- int main ()
 - Главная функция программы, запускает сервер

4.5.1 Подробное описание

Главный файл программы.

Этот файл содержит функцию main, которая запускает сервер.

Дата

19.12.2024

Версия

1.0

Автор

Чумазин Г.К.

4.6 server.h

```
1 #ifndef SERVER_H
2 #define SERVER_H
3
4 #include <string>
5 #include "logger.h"
6 #include "interface.h"
7 #include "calc.h"
8 /*
9  * @brief Класс server основной цикл для обработки сетевых операций. Запускает сервер
10  */
11 class Server {
12 public:
13     Server(const std::string& configFile, int port, Logger& logger);
14     void start();
15
16 private:
17     std::string configFile;
18     int port;
19     Logger& logger;
20 };
21
22 #endif // SERVER_H
```


Предметный указатель

Auth, [5](#)

Calculator, [5](#)

CommandLineInterface, [6](#)
 printHelp, [6](#)

Logger, [7](#)

printHelp
 CommandLineInterface, [6](#)

Server, [7](#)

server.cpp, [10](#)