

TCPServer

Создано системой Doxygen 1.9.1

1	Алфавитный указатель классов	1
1.1	Классы	1
2	Список файлов	3
2.1	Файлы	3
3	Классы	5
3.1	Класс DataBaseLoader	5
3.1.1	Подробное описание	5
3.1.2	Методы	5
3.1.2.1	checkUserLogin()	6
3.1.2.2	getUser()	6
3.1.2.3	getUserPassword()	6
3.1.2.4	loadData()	6
3.1.3	Данные класса	6
3.1.3.1	delimiter	6
3.1.3.2	filePath	6
3.1.3.3	users	7
3.2	Класс Logger	7
3.2.1	Методы	7
3.2.1.1	WriteError()	7
3.2.2	Данные класса	7
3.2.2.1	filePath	7
3.3	Класс Opts	8
3.3.1	Подробное описание	8
3.3.2	Методы	8
3.3.2.1	usage()	8
3.3.3	Данные класса	8
3.3.3.1	confPath	8
3.3.3.2	logPath	9
3.3.3.3	port	9
3.4	Класс TCPServer	9
3.4.1	Конструктор(ы)	10
3.4.1.1	TCPServer()	10
3.4.1.2	~TCPServer()	10
3.4.2	Методы	10
3.4.2.1	calcFunction()	10
3.4.2.2	closeServer()	10
3.4.2.3	handle()	11
3.4.2.4	startServer()	11
3.5	Структура User	11
3.5.1	Подробное описание	11
4	Файлы	13

4.1 Файл TCPServer.cpp	13
4.1.1 Подробное описание	14
Предметный указатель	15

Глава 1

Алфавитный указатель классов

1.1 Классы

Классы с их кратким описанием.

DataBaseLoader	5
Logger	7
Opts	8
TCPServer	9
User	11

Глава 2

Список файлов

2.1 Файлы

Полный список документированных файлов.

DataBaseLoader.h	??
Logger.h	??
TCPServer.cpp	13
TCPServer.h	??

Глава 3

Классы

3.1 Класс DataBaseLoader

```
#include <DataBaseLoader.h>
```

Открытые члены

- DataBaseLoader (std::string newfilePath)
- [User](#) [getUser](#) (std::string i)
- bool [checkUserLogin](#) (std::string login)
- std::string [getUserPassword](#) (std::string login)

Закрытые члены

- void [loadData](#) ()

Закрытые данные

- std::string [filePath](#)
- std::map< std::string, [User](#) > [users](#)
- std::string [delimiter](#) = " "

3.1.1 Подробное описание

Класс для доступа к данным о пользователях

3.1.2 Методы

3.1.2.1 checkUserLogin()

```
bool DataBaseLoader::checkUserLogin (
    std::string checkLogin )
```

Проверить логин пользователя

Проверить наличие пользователя

3.1.2.2 getUser()

```
User DataBaseLoader::getUser (
    std::string i )
```

Вернуть данные о пользователе по ID

Вернуть пользователя по ID

3.1.2.3 getUserPassword()

```
std::string DataBaseLoader::getUserPassword (
    std::string checkLogin )
```

Вернуть пароль пользователя

3.1.2.4 loadData()

```
void DataBaseLoader::loadData ( ) [private]
```

Загрузить базу из файла

Считать файл с пользователями в переменную кеша

3.1.3 Данные класса

3.1.3.1 delimiter

```
std::string DataBaseLoader::delimiter = " " [private]
```

Разделитель строки в файле

3.1.3.2 filePath

```
std::string DataBaseLoader::filePath [private]
```

Путь к файлу

3.1.3.3 users

```
std::map<std::string, User> DataBaseLoader::users [private]
```

закешированная база пользователей

Объявления и описания членов классов находятся в файлах:

- DataBaseLoader.h
- DataBaseLoader.cpp

3.2 Класс Logger

Открытые члены

- Logger (std::string filePath)
- void WriteError (std::string error, int criticalCode)

Закрытые данные

- std::string filePath

3.2.1 Методы

3.2.1.1 WriteError()

```
void Logger::WriteError (  
    std::string error,  
    int criticalCode )
```

Создать запись в логах

Создание записи в логах ошибок

3.2.2 Данные класса

3.2.2.1 filePath

```
std::string Logger::filePath [private]
```

Путь к файлу

Объявления и описания членов классов находятся в файлах:

- Logger.h
- Logger.cpp

3.3 Класс Opts

Открытые члены

- Opts (int argc, char **argv)
- std::string getLogPath ()
- std::string getConPath ()
- int getPort ()

Закрытые члены

- void usage (const char *progName)

Закрытые данные

- std::string logPath = "/var/log/vcalc.log"
- std::string confPath = "/etc/vcalc.conf"
- int port = 33333

3.3.1 Подробное описание

Аргументы коандной строки

3.3.2 Методы

3.3.2.1 usage()

```
void Opts::usage (  
    const char * progName ) [private]
```

Вывод подсказки

3.3.3 Данные класса

3.3.3.1 confPath

```
std::string Opts::confPath = "/etc/vcalc.conf" [private]
```

Путь к файлу с данными

3.3.3.2 logPath

```
std::string Opts::logPath = "/var/log/vcalc.log" [private]
```

Путь к файлу с логами

3.3.3.3 port

```
int Opts::port = 33333 [private]
```

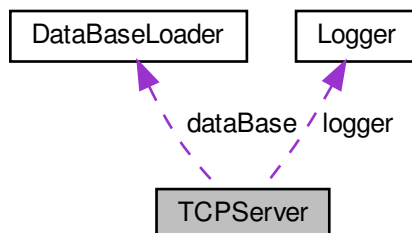
Номер порта

Объявления и описания членов класса находятся в файле:

- TCPSetverProj.cpp

3.4 Класс TCPServer

Граф связей класса TCPServer:



Открытые члены

- [TCPServer](#) (int newPort, std::string logPath, std::string confPath)
- [~TCPServer](#) ()
- void [startServer](#) ()
- void [closeServer](#) ()
- void [handle](#) ()

Закрытые члены

- long long [calcFunction](#) (std::vector< long long > vector)
- char * [recvData](#) (int socket)
- void [sendData](#) (int socket, void *data)

Закрытые данные

- unsigned short port
- int this_s
- [Logger](#) logger
- [DataBaseLoader](#) dataBase
- std::string errorMsg = "ERR"
- std::string okClientMsg = "OK"

3.4.1 Конструктор(ы)

3.4.1.1 TCPServer()

```
TCPServer::TCPServer (
    int newPort = 33333,
    std::string logPath = "/var/log/vcalc.log",
    std::string confPath = "/etc/vcalc.conf" )
```

Реализация конструктора

3.4.1.2 ~TCPServer()

```
TCPServer::~~TCPServer ( )
```

Реализация деструктора

3.4.2 Методы

3.4.2.1 calcFunction()

```
long long TCPServer::calcFunction (
    std::vector< long long > vector ) [private]
```

Функция обработки векторов

3.4.2.2 closeServer()

```
void TCPServer::closeServer ( )
```

реализация закрытие сервера

3.4.2.3 handle()

```
void TCPServer::handle ( )
```

реализация основной логики

3.4.2.4 startServer()

```
void TCPServer::startServer ( )
```

реализация запуска сервера

Объявления и описания членов классов находятся в файлах:

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

3.5 Структура User

```
#include <DataBaseLoader.h>
```

Открытые атрибуты

- std::string login
- std::string password

3.5.1 Подробное описание

Структура данных о пользователях

Объявления и описания членов структуры находятся в файле:

- DataBaseLoader.h

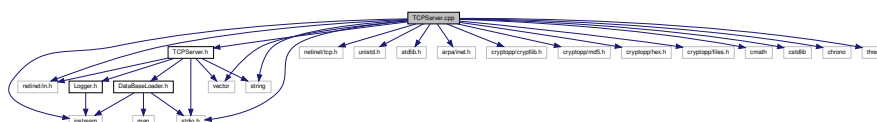
Глава 4

Файлы

4.1 Файл TCPServer.cpp

```
#include "TCPServer.h"  
#include <netinet/in.h>  
#include <netinet/tcp.h>  
#include <unistd.h>  
#include <stdio.h>  
#include <stdlib.h>  
#include <string>  
#include <vector>  
#include <arpa/inet.h>  
#include <cryptopp/cryptlib.h>  
#include <cryptopp/md5.h>  
#include <cryptopp/hex.h>  
#include <cryptopp/files.h>  
#include <iostream>  
#include <cmath>  
#include <cstdlib>  
#include <chrono>  
#include <thread>
```

Граф включаемых заголовочных файлов для TCPServer.cpp:



Макросы

- `#define cplusplus`
- `#define CRYPTOPP_ENABLE_NAMESPACE_WEAK 1`

4.1.1 Подробное описание

Версия

1.0

Авторство

ИБСТ ПГУ

Предупреждения

учебный пример

Предметный указатель

- ~TCPServer
 - TCPServer, [10](#)
- calcFunction
 - TCPServer, [10](#)
- checkUserLogin
 - DataBaseLoader, [5](#)
- closeServer
 - TCPServer, [10](#)
- confPath
 - Opts, [8](#)
- DataBaseLoader, [5](#)
 - checkUserLogin, [5](#)
 - delimiter, [6](#)
 - filePath, [6](#)
 - getUser, [6](#)
 - getUserPassword, [6](#)
 - loadData, [6](#)
 - users, [6](#)
- delimiter
 - DataBaseLoader, [6](#)
- filePath
 - DataBaseLoader, [6](#)
 - Logger, [7](#)
- getUser
 - DataBaseLoader, [6](#)
- getUserPassword
 - DataBaseLoader, [6](#)
- handle
 - TCPServer, [10](#)
- loadData
 - DataBaseLoader, [6](#)
- Logger, [7](#)
 - filePath, [7](#)
 - WriteError, [7](#)
- logPath
 - Opts, [8](#)
- Opts, [8](#)
 - confPath, [8](#)
 - logPath, [8](#)
 - port, [9](#)
 - usage, [8](#)
- port
 - Opts, [9](#)
- startServer
 - TCPServer, [11](#)
- TCPServer, [9](#)
 - ~TCPServer, [10](#)
 - calcFunction, [10](#)
 - closeServer, [10](#)
 - handle, [10](#)
 - startServer, [11](#)
 - TCPServer, [10](#)
 - TCPServer.cpp, [13](#)
- usage
 - Opts, [8](#)
- User, [11](#)
- users
 - DataBaseLoader, [6](#)
- WriteError
 - Logger, [7](#)