SERVER_KURSOVAYA

Создано системой Doxygen 1.9.8

Глава 1

Алфавитный указатель классов

1.1 Классы

Классы с их кратким описанием.

Глава 2

Список файлов

2.1 Файлы

Полный список документированных файлов.

Глава 3

Классы

3.1 Класс Calculator

Класс для выполнения вычислений с векторами, полученными от клиента.

Открытые члены

- uint16_t processVectors (int socket)
  Обрабатывает векторы, полученные от клиента.

3.1.1 Подробное описание

Класс для выполнения вычислений с векторами, полученными от клиента.

3.1.2 Методы

3.1.2.1 processVectors()

uint16_t Calculator::processVectors (
int socket) [inline]

Обрабатывает векторы, полученные от клиента.

Аргументы

socket	Дескриптор сокета, через который происходит взаимодействие с клиентом.

Возвращает

0 в случае успешной обработки, -1 в случае ошибки.

Объявления и описания членов класса находятся в файле:

- server.cpp

3.2 Класс ClientCommunicate

Класс для обработки общения с клиентом.

Открытые члены

- void communicate (int socket, const std::string &userDbFileName, const std::string &logFileName)
  Обрабатывает запросы клиента, включая аутентификацию и вычисления.

3.2.1 Подробное описание

Класс для обработки общения с клиентом.

3.2.2 Методы

3.2.2.1 communicate()

```
void ClientCommunicate::communicate (
                int socket,
                const std::string & userDbFileName,
                const std::string & logFileName )   [inline]
```

Обрабатывает запросы клиента, включая аутентификацию и вычисления.

Аргументы

socket	Дескриптор сокета для взаимодействия с клиентом.
userDbFileName	Имя файла базы данных пользователей.
logFileName	Имя файла для логирования.

Объявления и описания членов класса находятся в файле:

- server.cpp

3.3 Класс ConnectToBase

Класс для аутентификации пользователей, используя данные из базы.

Открытые члены

- bool authenticateUser (const std::string &login, const std::string &salt, const std::string &clientHash, const std::string &dbFileName)
  Аутентифицирует пользователя, проверяя логин и хэш пароля.

Закрытые члены

- std::string hashPassword (const std::string &password, const std::string &salt)

  Хэширует пароль с использованием соли.
- bool compareHashes (const std::string &serverHash, const std::string &clientHash)

  Сравнивает два хэша паролей.

3.3.1 Подробное описание

Класс для аутентификации пользователей, используя данные из базы.

3.3.2 Методы

3.3.2.1 authenticateUser()

```
bool ConnectToBase::authenticateUser (
                const std::string & login,
                const std::string & salt,
                const std::string & clientHash,
                const std::string & dbFileName )    [inline]
```

Аутентифицирует пользователя, проверяя логин и хэш пароля.

Аргументы

login	Логин пользователя.
salt	Соль, используемая для хэширования пароля.
clientHash	Хэш пароля, присланный клиентом.
dbFileName	Имя файла базы данных пользователей.

Возвращает

true, если аутентификация успешна; false в противном случае.

3.3.2.2 compareHashes()

```
bool ConnectToBase::compareHashes (
                const std::string & serverHash,
                const std::string & clientHash )    [inline], [private]
```

Сравнивает два хэша паролей.

Аргументы

serverHash	Хэш, полученный сервером.
clientHash	Хэш, полученный от клиента.

Возвращает

true, если хэши совпадают; false в противном случае.

3.3.2.3 hashPassword()

std::string ConnectToBase::hashPassword (
const std::string & password,
const std::string & salt) [inline], [private]

Хэширует пароль с использованием соли.

Аргументы

password	Пароль пользователя.
salt	Соль.

Возвращает

Хэшированный пароль в виде строки.

Объявления и описания членов класса находятся в файле:

- server.cpp

3.4 Класс Error

Класс для обработки ошибок в приложении.

Открытые статические члены

- static void logError (const std::string &message, bool isCritical=false)
  Логирует ошибку в консоль.

3.4.1 Подробное описание

Класс для обработки ошибок в приложении.

3.4.2 Методы

3.4.2.1 logError()

static void Error::logError (
const std::string & message,
bool isCritical = false) [inline], [static]

Логирует ошибку в консоль.

Аргументы

message	Сообщение об ошибке.
isCritical	Указывает, является ли ошибка критической.

Объявления и описания членов класса находятся в файле:

- server.cpp

3.5 Класс Interface

Класс для вывода информации о правильном использовании программы и логирования.

Открытые статические члены

- static void printUsage ()
  Выводит информацию о правильном использовании программы.
- static void logError (const std::string &logFileName, const std::string &message, bool isCritical)
  Логирует ошибку в файл.
- static void logMessage (const std::string &logFileName, const std::string &message)
  Логирует информационное сообщение в файл.

3.5.1 Подробное описание

Класс для вывода информации о правильном использовании программы и логирования.

3.5.2 Методы

3.5.2.1 logError()

```
static void Interface::logError (
            const std::string & logFileName,
            const std::string & message,
            bool isCritical )    [inline], [static]
```

Логирует ошибку в файл.

Аргументы

logFileName	Имя лог-файла.
message	Сообщение об ошибке.
isCritical	Указывает, является ли ошибка критической.

3.5.2.2 logMessage()

static void Interface::logMessage (
const std::string & logFileName,
const std::string & message) [inline], [static]

Логирует информационное сообщение в файл.

Аргументы

logFileName	Имя лог-файла.
message	Информационное сообщение.

Объявления и описания членов класса находятся в файле:

- server.cpp

Глава 4

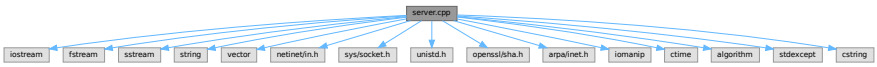
Файлы

4.1 Файл server.cpp

Сервер курсовой работы.

```
#include <iostream>
#include <fstream>
#include <sstream>
#include <string>
#include <vector>
#include <netinet/in.h>
#include <sys/socket.h>
#include <unistd.h>
#include <openssl/sha.h>
#include <arpa/inet.h>
#include <iomanip>
#include <ctime>
#include <algorithm>
#include <stdexcept>
#include <cstring>
```

Граф включаемых заголовочных файлов для server.cpp:



Классы

- class Error

  Класс для обработки ошибок в приложении.
- class Calculator

  Класс для выполнения вычислений с векторами, полученными от клиента.
- class ConnectToBase

  Класс для аутентификации пользователей, используя данные из базы.
- class Interface

  Класс для вывода информации о правильном использовании программы и логирования.
- class ClientCommunicate

  Класс для обработки общения с клиентом.

Функции

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

  Основная программа сервера, принимающая подключения и обрабатывающая запросы клиентов.

4.1.1 Подробное описание

Сервер курсовой работы.

Серверная часть программы, предназначенная для:

- Аутентификации пользователей на основе логина и пароля.
- Получения векторов от клиента и выполнения над ними вычислений.
- Логирования операций и ошибок.

Автор

Бренинг И. А.