Техническое задание

1. Введение

Документ описывает требования к разработке многопоточного сервера для обработки произвольных файлов. Сервер будет взаимодействовать с клиентской программой по сети TCP/IP.

2. Описание функциональности сервера

Сервер должен поддерживать следующие операции:

1) Загрузка файла на сервер.

2) Получение файла с сервера по его имени или идентификатору.

3) Удаление файла с сервера по его имени или идентификатору.

4) Завершение работы сервера по запросу клиента.

3. Обработка запросов клиента

1) Сервер должен ожидать подключения клиентов по сети TCP/IP.

2) После подключения клиента сервер должен принимать команды от него и выполнять соответствующие действия

3) Для каждого подключения должен создаваться отдельный поток для обработки запросов клиента.

4. Операции сервера

1) Загрузка файла на сервер:

- Клиент отправляет команду "1".

- Сервер получает имя файла и его содержимое от клиента.

- Сервер сохраняет файл в своей директории, генерируя уникальный идентификатор для него.

- Сервер отвечает клиенту кодом состояния и, в случае успешного сохранения, возвращает идентификатор файла.

2) Получение файла с сервера:

- Клиент отправляет команду "2".

- Клиент указывает способ получения файла (по имени или по идентификатору).

- Сервер отправляет файл клиенту, если он существует, иначе возвращает код ошибки.

3) Удаление файла с сервера:

- Клиент отправляет команду "3".

- Клиент указывает способ удаления файла (по имени или по идентификатору).

- Сервер удаляет файл с сервера и возвращает код состояния.

4) Завершение работы сервера:

- Клиент отправляет команду "exit".

- Сервер завершает свою работу.

5. Требования к хранению информации

Информация о файлах и их идентификаторах должна храниться в памяти сервера и сохраняться в файл при завершении работы сервера.

6. Требования к обработке ошибок

Сервер должен корректно обрабатывать все возможные ошибки во время выполнения операций (например, отсутствие файла, недопустимые команды и т. д.).

7. Требования к безопасности

1) Сервер должен обеспечивать безопасное сохранение и передачу файлов между клиентом и сервером.

2) Сервер должен быть защищен от нежелательного доступа и злоумышленных атак.

8. Требования к производительности

Сервер должен обрабатывать запросы клиентов эффективно и быстро, даже при одновременном обращении нескольких клиентов.

9. Завершение работы сервера

По завершении работы сервера, информация о файлах и их идентификаторах должна сохраняться в файл для последующего использования при перезапуске.