

Разработать серверное и клиентское кроссплатформенные приложения на C++.

Клиент: консольное приложение, работающее в двух режимах:

- При запуске с параметрами подключения `host:port` (`./Console host:port`) подключается к серверу и отправляет пакет с данными через случайный интервал времени 5-30 секунд. Соединение с сервером должно постоянно поддерживаться.
- При запуске с параметрами подключения + флаг (`./Console host:port --statistic`) подключается к серверу, запрашивает статистику, выводит на консоль и завершает работу.

Сервер: консольное приложение, ожидающее **WebSocket** подключение на указанном в консольном параметре порту. При получении пакета данных сохраняет его в **БД SQLite**. При получении запроса статистики отправляет результат клиенту. Поддерживает одновременное подключение нескольких клиентов.

Данные запросов/ответов нужно сериализовывать в **protobuf**. Пакет с данными содержит уникальный идентификатор (`uuid`) клиента, временную метку на стороне клиента в таймзоне UTC и два случайных вещественных числа `x`, `y` в интервале `[-90, 90]`.

Статистика содержит в себе **среднее значение числа `x`** и **сумму значений числа `y` по модулю** за интервал 1 минута, 5 минут в разрезе каждого клиента.

Пример вывода статистики:

UUID X_1 Y_1 X_5 Y_5

f7395d89-1259-429a-a1c0-a5d3f1b13096 23.44 2356 16.76 6678.5

b8f1b880-0b91-4d0a-857c-3c8c4e887ff9 11.22 1589 -11.76 4682.75

При разработке нужно придерживаться стандартов C++11/14/17, использования стандартной библиотеки, чистоты и декомпозиции кода. Приветствуется использование библиотеки **boost**.

Проект может быть под MSVC или Clang или иное, главное чтобы был кроссплатформенный сам код.