C# programmer assignment

Create a utility using provided MT5 API libs. These libraries serve as a .NET wrapper above the MT5 trading server API. All necessary API functions documentation is provided as well.
We want you to create *watchdog* utility, which will monitor trading activity in real-time and will log any suspicious behavior.
Keep in your mind that:

- the utility should be designed so that it can be easily maintained and easily extended for other functionality,
- there can be a lot of real-time data incoming in a short period of time,
- you should use available libraries reasonably (.net, nugget),
- provided libraries (MT5) should be distributed together with the app,
- the solution should be provided in the form of an online git repository.

Watchdog

Trading servers can have some restrictions, some users can try to bypass these restrictions by using multiple accounts with different names (of some relatives for example) and on multiple servers. We need to detect such behavior in real-time. Check new deals and if two or more similar deals exist, log the information. Similar deals are deals where open time differs no more than one second, currency pair is the same and the difference in volume-to-balance ratio is no more than 5%. If such deals are detected log info including accounts and servers where the deals were detected.

The utility should be configured using params or config file. Configuration params should contain:

- open time delta,
- trade volume to balance ratio,
- servers to connect (multiple servers can be provided at once).

Notes:

- in all cases only buy/sell deals are relevant (see the provided documentation),
- application should be able to compare data from multiple servers (cross server detection of similar deals is necessary),
- incoming trade records can be compared in *one-to-many* principle,
- volume to balance ratio is ratio between the deal volume and current user balance, user balance should be requested during every comparison as it changes in time

Example 1

List of incoming deals:

```
Deal #1, Balance 10 000, Buy EURUSD 1 lot at 2019-05-12 14:43:12
Deal #2, Balance 10 000, Sell GBPUSD 0.2 lots at 2019-05-12 14:43:23
Deal #3, Balance 10 000, Sell GBPUSD 0.21 lot at 2019-05-12 14:43:24 <- triggered
match with deal #2
```

Example 2

List of incoming deals:

```
Deal #1, Balance 10 000, Buy EURUSD 1 lot at 2019-05-12 14:43:12
Deal #2, Balance 10 000, Sell GBPUSD 0.2 lots at 2019-05-12 14:43:23
Deal #3, Balance 1 000, Sell GBPUSD 1.2 lots at 2019-05-12 14:43:23
Deal #4, Balance 10 000, Sell GBPUSD 0.21 lot at 2019-05-12 14:43:24 <- triggered
match with deal #2
Deal #5, Balance 20 000, Sell GBPUSD 0.4 lot at 2019-05-12 14:43:24 <- triggered match
with deal #2 and deal #4
```