

Задача

Создать API сервер на ASP.NET которое будет симулировать очередь задач. Это необходимо для организации работы с ресурсом, который может исполнять операции только синхронно. Для симуляции синхронной операции будет использоваться запись в файл. Структура работы API:

1. Сервер получает запрос
2. Запрос отправляется в очередь исполнения
3. Для каждого запроса в очереди:
 1. Открывается файл записи
 2. Производится запись в файл данных запроса (формат указан ниже)
 3. Файл закрывается и сервис ждет 50-100 миллисекунд (для симуляции дорогой операции, которая должна обрабатываться синхронно)
 4. Отправляется ответ с данными записи в файл

Требования к решению

Использовать только стандартную библиотеку для имплементации решения, кроме классов, которые могут выполнять основной функционал решения (например, функционал очереди). В решении должен присутствовать синглтон-сервис, который реализует полный функционал очереди исполнения, то есть получения асинхронных запросов на исполнение по одному, их обработка и отправка ответа на запрос. Способ реализации очереди можете выбрать сами.

Формат запроса

API имеет только один входящий запрос со следующим форматом:

```
POST /api/process
Content-Type: application/json

{
    "message": "string"
}
```

Поле `message` является произвольной строкой до 100 символов, которая будет записываться в файл. При несоответствии запроса формату, должен приходить ответ `HTTP 400 Bad Request`

Формат записи в файл

Каждый запрос будет записываться в файл с следующим формате:

```
<время получения запроса> | <время начала записи в файл> | <message>
```

- `Время получения запроса` фиксируется в момент получения запроса, перед отправкой в очередь исполнения
- `Время начала записи в файл` фиксируется перед записью в файл

Время записывается в формате `ISO 8601` (Например, 2024-04-14T17:56:03)

Формат ответа

Когда данные запроса записываются в файл, должен прийти ответ следующего формата

```
{
    // Время получения запроса
    "requestTime": "ISO 8601 datetime string",

    // Время начала записи в файл
    "writeTime": "ISO 8601 datetime string",

    // Количество миллисекунд, за которое запрос обработался
    (writeTime - requestTime)
    "processingTime": "number"
}
```

Бонус

Сделать обработку нескольких запросов из очереди одновременной - если в начале обработки следующего запроса в очереди присутствуют несколько запросов, то вместо обработки по одному, собрать до 5 запросов и обработать одновременно:

- Если в очереди присутствует больше 5 запросов, взять первые 5 и обработать остальные на следующей итерации
- Если в очереди присутствует меньше 5 запросов, то взять обработать все запросы.