Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение

высшего профессионального образования

«Владимирский государственный университет

имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем и программной инженерии

Лабораторная работа №6

по дисциплине "Технологии программирования"

ТЕМА РАБОТЫ:

Создание и работа с внешними библиотеками, подключаемыми к .NET приложений (на примере библиотеки логирования)

Выполнил:

студент гр. ПРИ-120

Парахин К.В.

Принял:

Ассистент кафедры ИСПИ

Данилов В.В.

Владимир 2022 г.

Цель работы:

Познакомиться с порядком подключения внешних библиотек к .NET приложению и их использования в нём.

Научиться работать с одной из библиотекой логирования в .NET (на примере библиотеки Serilog)

Выполнение работы:

Начнем работу с установки требуемых NuGet пакетов с файлами библиотеки Serilog, Serilog.AspNetCore и Serilog.Slinks.Debug

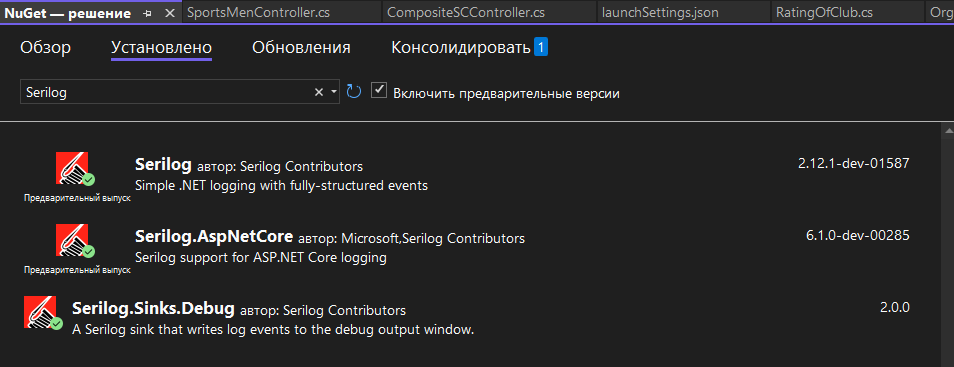


Рисунок 1. Установка NuGet пакетов

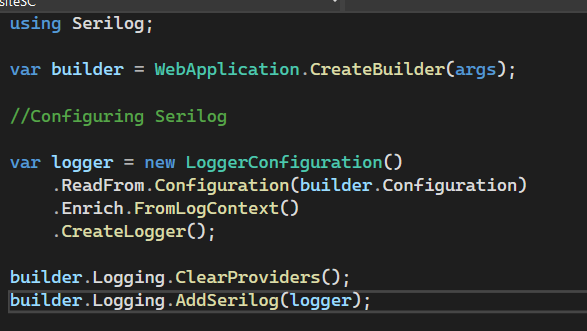


Рисунок 2. Добавление конфигурации Serilog в Program.cs композитного микросервиса

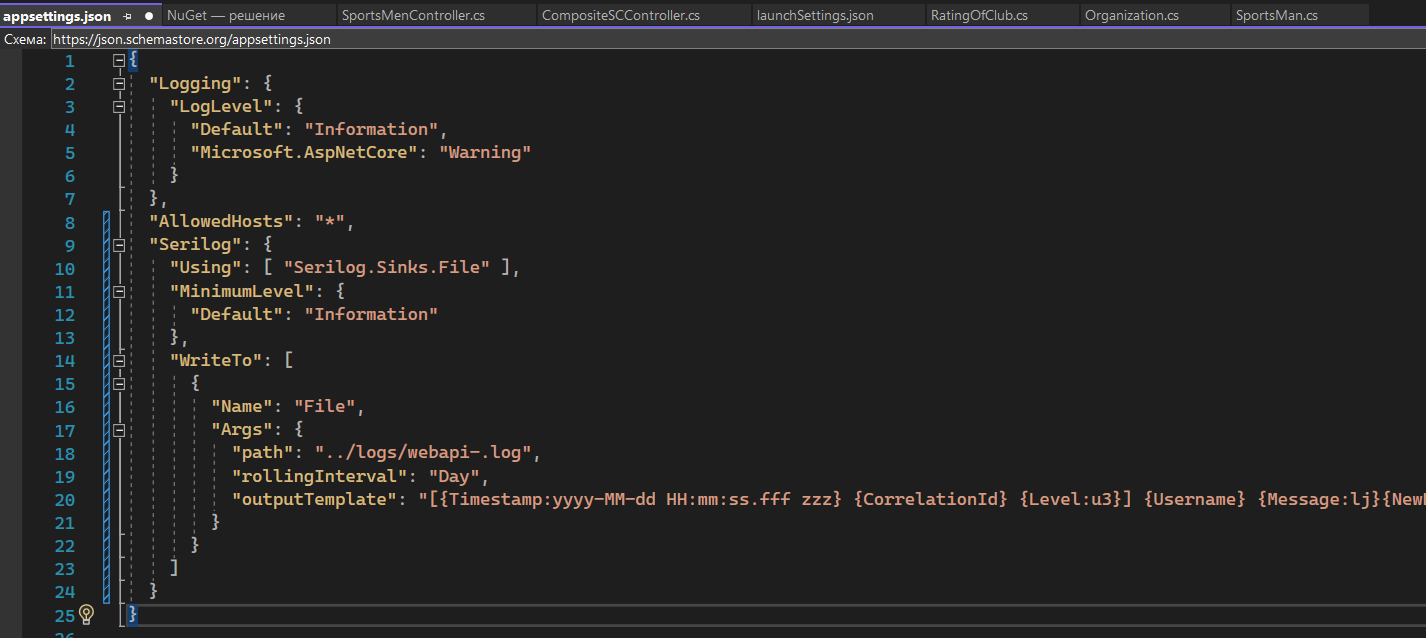


Рисунок 3. Добавление конфигурации Serilog в файл appsettings.json

Внутри контроллера CompositeSCController добавим переменную для логирования и в GET – метод добавим строку с выводом информации из объекта логера:

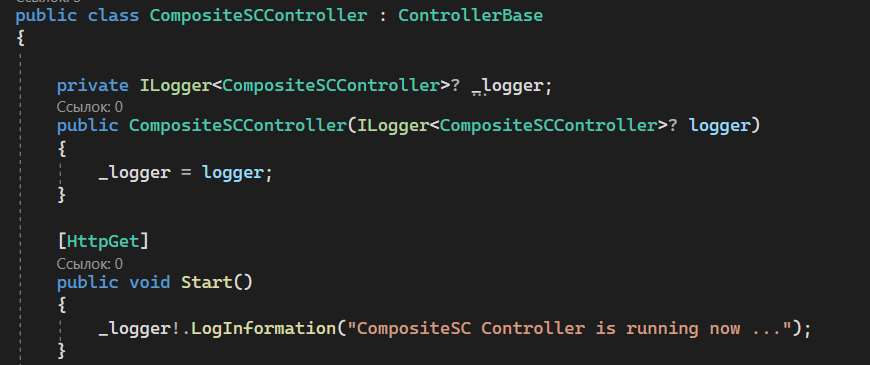


Рисунок 4. Добавление метода работы с логером

Далее добавим остальные методы работы с логером в остальные методы нашего контроллера, проведем сборку и запуск веб-приложения, основанного на функционировании двух микросервисов внутри композитного микросервиса.

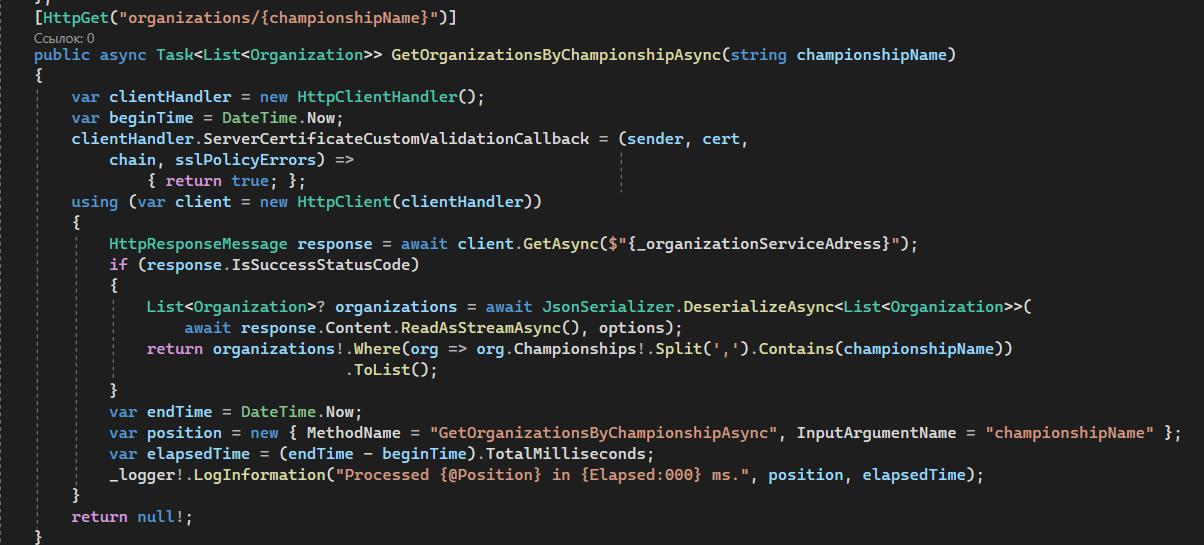


Рисунок 5. Логер для метода GetOrganizationByChampionshipAsync

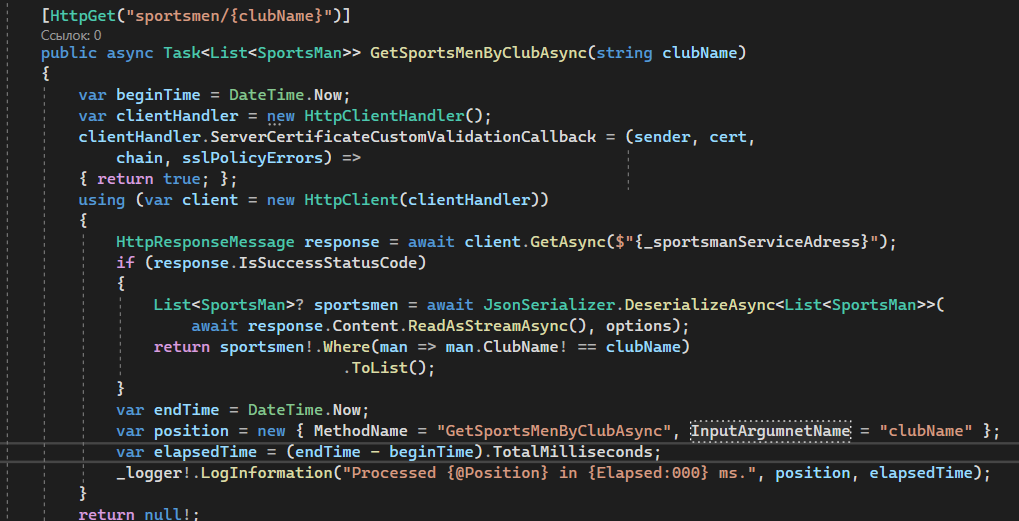


Рисунок 6. Логер для метода GetSportsmenByClubAsync

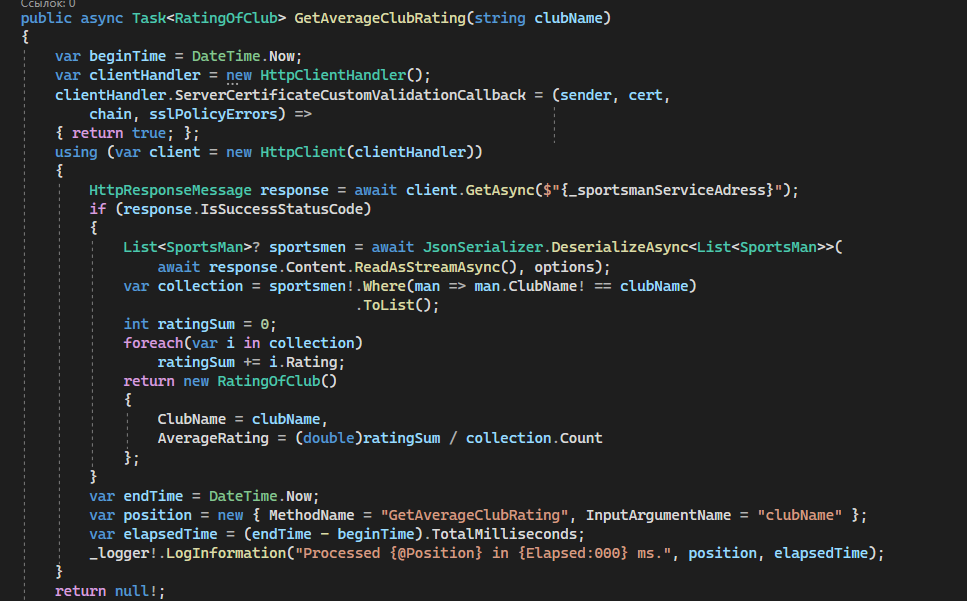


Рисунок 7. Логер для метода GetAverageClubRating

А затем просмотрим появившиеся после запуска приложения файлы с расширением .log внутри корневого каталога проекта:

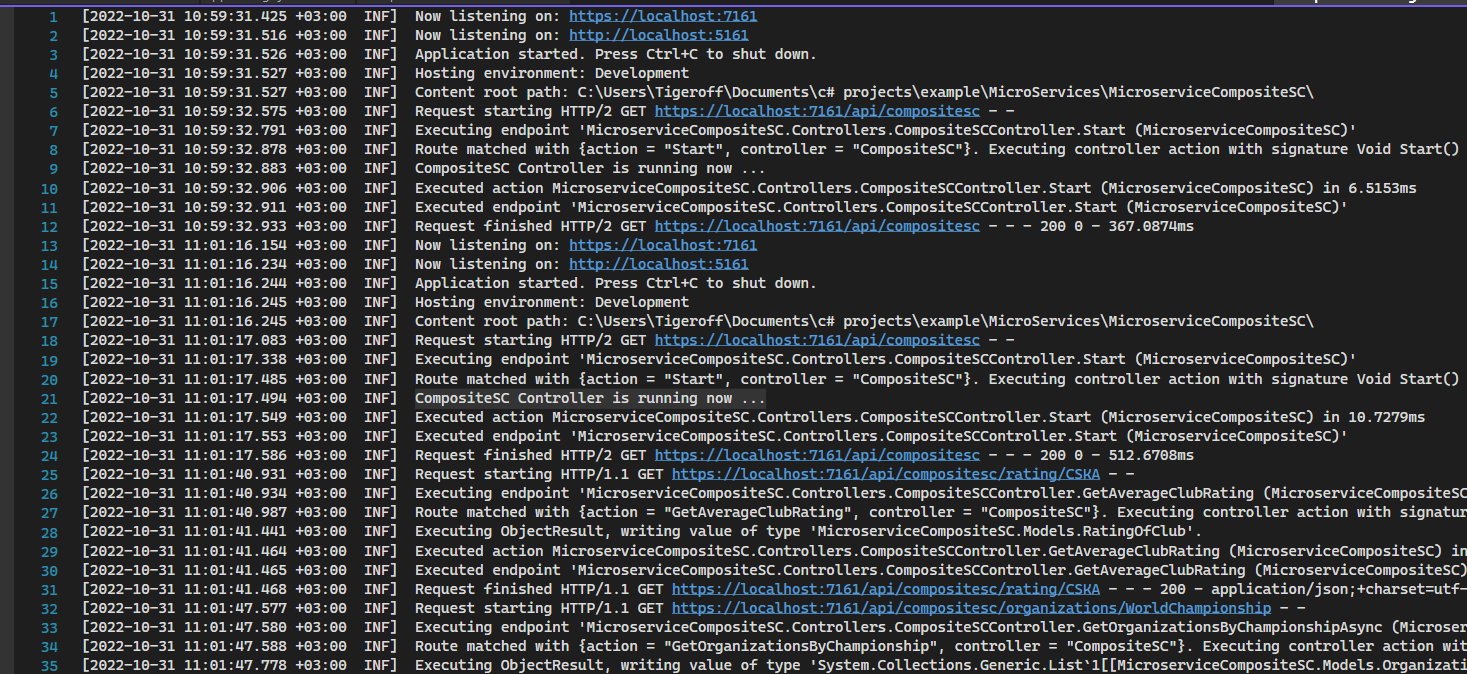


Рисунок 8. Логи веб-приложения

Затем создадим еще один проект (по шаблону «Консольное приложение») в этом же решении и добавим туда класс, осуществляющий вызов требуемого .dll файла сборки и вывод логов работы очередей (из лабораторной работы №4).

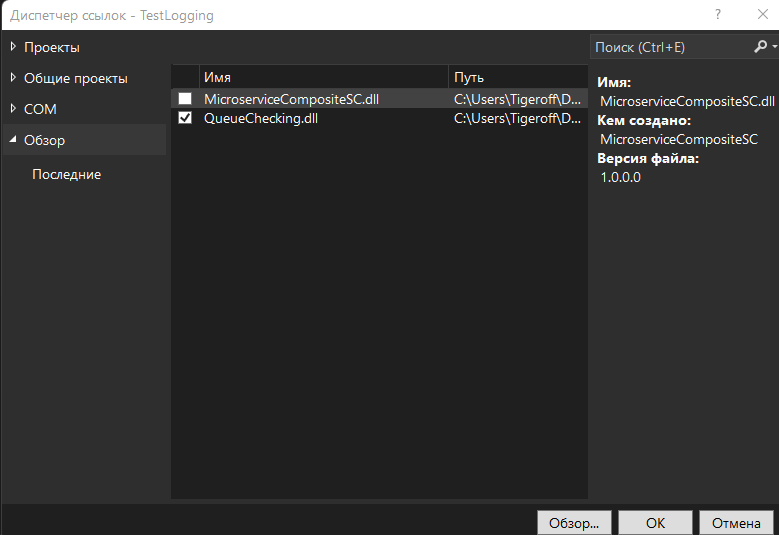


Рисунок 9. Подключение QueueChecking.dll к текущему проекту

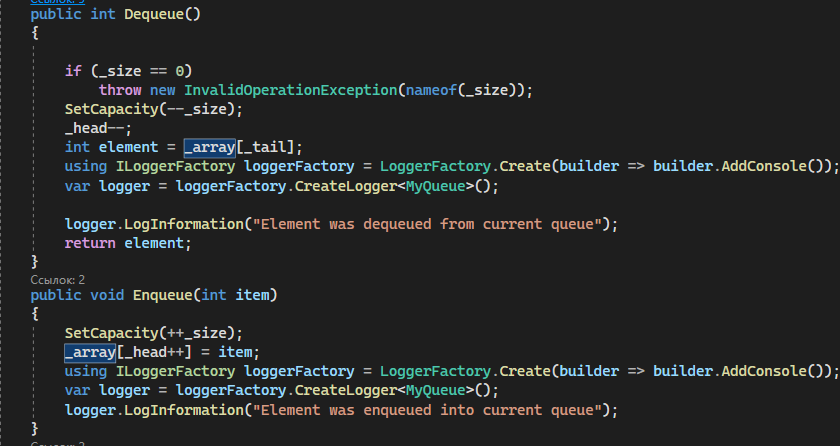


Рисунок 10. Добавление логирования в консоль в методах класса Queue проекта QueueChecking.dll

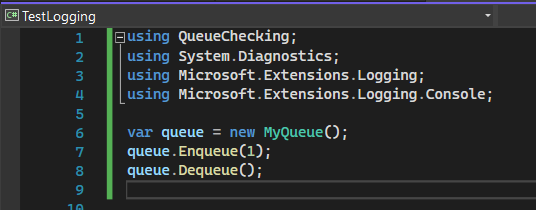


Рисунок 11. Добавление вызова методов в проекте TestLogging из подключаемой библиотеки QueueChecking.dll

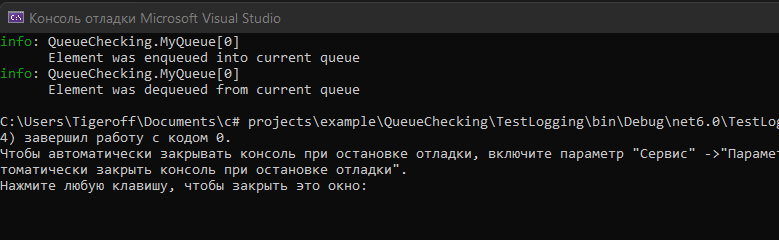


Рисунок 12. Вывод логов в консоль

Вывод

В результате выполнения работы, я научился работать с логированием в .NET, используя бибилиотеку Serilog.