Служебная записка о технологиях Microsoft (предварительная версия)

инженер-программист Нижегородов О.Э.

28.12.2023

Поскольку я работаю с технологией NetCore 3.1 я могу уже многое рассказать о эффективности применения данной технологии для интеграции с такими системами, как региональная система .. Несмотря на то, что программа направлений сделана в форме настольного приложения, уже в ней заложены многие возможности, которые эффективно могут быть использованы, а также дальнейшие возможности развития данного момента.

Итак.

1. Асинхронная обработка процессов.

Такая современная и актуальная обработка в десктопном приложении, как асинхронность в веббраузере, применима как стандарт работы с возможно долгими процессами (получение данных по вебсервисам, загрузка данных из БД и т.п.). Это процессы, которые не блокируют например в Windows графический интерфейс, поскольку он сильно связан с операционной средой. Платформа NetCore такие процессы запускает в форме задачи и ожидает ее завершения, не блокируя другие процессы. Таким образом можно использовать это для планирования задач, которые будут исполняться по плану, минуя ручной труд человека. Это задача приближения к полной автоматизации работы с информацией.

2. Сериализация-Десериализация объектов.

Сериализация объекта используется для конвертирования информационного объекта в поток, который может быть послан эффективно по сети. Десериализация используется для обратного конвертирования потока в информационные объекты. Именно так происходит при работе с вебсервисами. В программе направлений достаточно хорошо зарекомендовала себя библиотека Newtonsoft. Она хорошо справляется с кириллицей.

В данном случае из региональной системы (PHP или JAVA платформа) посылает сообщение и настольная программа успешно принимает это информационное сообщение.

Объктивности ради, хочу заметить, что пплатфома NetCore сделана по прототипу JAVA технологий, развивающихся в свое время в сегменте коммунити. NetCore - это развитие NetFrameworka (прототип JAVA технологий), работающего на ОС Windows и вобравшего в себя все лучшее из одной слаженной команды Microsoft. Это в первую очередь фреймворк, заточенный под взаимодействие пролцессов и дающий много различных методов, ускоряющих рахработку приложений. На более глубокие знания по NetCore у меня не было возможностей из-за других задач, но уже сейчас можно использовать средство для создания баз данных по моделям - это Entity Framework - и такая технология используется в JAVA а также в PHP и др. В частности я провел эксперименты с БД Постгрес 10 успешно.  
Постгрес начиная с 10 версии позволяет работать с автоинкрементом ключевого поля наподобие MSSQL. Также были проведены эксперименты с веб приложением, представляющим из себя набор веб страниц шаблонизатора Razor (RazorPages). При создании такого приложения используется так называемый Scaffolding процесс - автогенерация страниц опять же по моделям. Все это может ускорить разработку приложения в целом а также уменьшает ручное рутинное программирование, делая акцент на бизнес-процессах.

Такая методика применима и например в PHP, но тут коммерческие решения. В отличие от этого среда Microsoft Visual Studio community (стандартное средство разработчика для NetCore) дает это бесплатно.

Поэтому я изучаю программную инженерию Net технологии в Skillfactory. Были проведены эксперименты по созданию CRUD-страниц - страницы для создания (Create), чтения (Read), изменения данных (Update), удаления (Delete) записи информации. И это решается как раз с помощью автогенерации кода по моделям.   
Модели - это объекты, состоящие из набора полей.

Microsoft Visual Studio community на борту имеет много стартапов - в том числе начало для разработки API. Экспериментировал также для отдачи Json формата. Кириллица при этом не ломается.

Итак, имеем - настольное приложение выпоняет роль пользователя вебсервисов региональной системы. Также веб приложение RazorPages для обработки информации в форме записей (CRUD-страниц). Если сюда добавить приложение API получим что называется fullstack разработку (приложение полного цикла). И это все на одной платформе. При всем при этом используется объектно-ориентированный подход, который позволяет манипулировать уже имеющимися объектами. Это делается опять же для убыстрения разработки, используя уже готовые объекты (что называется используя уже написанный код).