

УДК 004.62

ИСПОЛЬЗОВАНИЕ ASP.NET MVC CORE ДЛЯ СОЗДАНИЯ ВЕБ-ПРИЛОЖЕНИЯ ПО КОНТРОЛЮ НАД ВЫПЛАТАМИ ПЕНСИЙ ДЛЯ ПЕНСИОННОГО ФОНДА РОССИЙСКОЙ ФЕДЕРАЦИИ

Трепаков М.В., студент гр. ИТб-171, IV курс
Сыркин И.С., к.т.н. доцент кафедры ИиАПС
Кузбасский государственный технический университет
имени Т. Ф. Горбачева
г. Кемерово

С наступлением пандемии во многих сферах начался переход персонала на удалённую работу. Коснулось это, в том числе, и ПФРФ. Текущий комплекс программ по контролю над выплатами пенсий не позволяет перевести работу в удалённый формат, так как работа с ним предполагается непосредственно на компьютерах ПФР. Но использование веб-интерфейса данной программы позволит сотрудникам работать даже из дома.

К тому же, сейчас рабочие места ПФР оснащены полноценными ПК с монитором и отдельным системным блоком для стабильной работы текущего комплекса программ, требующего относительно мощный ПК. При дальнейшем обновлении оборудования потребуется закупать такие же ПК, что отрицательно сказывается на бюджете, выделяемом для обновления базы технического оснащения ПФР.

Если же использовать веб-интерфейс, расположенный на сервере, то рабочие места можно будет оснастить моноблоками или же мини-компьютерами, что значительно уменьшит расход бюджета на обновление оборудования.

ASP.Net MVC Core – это среда разработки, имеющая открытый код и созданная компанией Microsoft, которую используют для создания Web-приложений, а также АПИ-интерфейсов. При этом используется специальная структура приложения под названием Model (Модель) View (Представление) Controller (Контроллер) (MVC).

Эта структура подразумевает деление разрабатываемого приложения на три основные части: модели, представления и контроллеры. Таким образом реализуется принцип разделения задач, который просто необходим при создании и поддержке приложений. Также, процесс обновления и доработки приложения становится в разы удобнее за счёт такого разделения.

Сама структура организована следующим образом: Пользовательский запрос отправляется непосредственно в контроллере (Controller), который его и обрабатывает. Затем контроллер взаимодействует с моделью (Model) и выполняет с ней действия, заданные пользователем.

После взаимодействия с моделью контроллер выбирает необходимое для отображения представление (View) со всеми данными из модели, которые необходимы пользователю.

Также, стоит отметить, что мы используем именно ASP.Net MVC Core, а не простой ASP.Net MVC, так как Core является мульти-платформенной средой разработки для Windows, MacOS и Linux, что позволяет устанавливать любую из этих операционных систем на рабочие компьютеры без ограничений.

В качестве используемой технологии для улучшения приложения Пенсионного Фонда Российской Федерации по контролю над выплатами пенсий мы используем именно технологию ASP.Net MVC Core, так как она позволяет разрабатывать веб-приложения, упрощающие установку, запуск и работу с такими программами.

При переходе в веб-интерфейс значительно упростится работа с программой, её установка и запуск. К тому же, офисы ПФР можно оборудовать моноблоками с упрощенными характеристиками и меньшей стоимостью в сравнении с полноценными персональными компьютерами.

Внедрение данной технологии в работу ПФР будет состоять из создания самого веб-приложения, расположения его на сервере, запуска и дальнейшей поддержки и обновления. Для пользователей переход текущего приложения на веб-версию не будет сопровождаться трудоёмким переобучением, так как общий интерфейс остаётся тем же. Персонал достаточно будет обучить процессу запуска веб-приложение (подключение к веб-сайту по определённой ссылке).

Принцип работы заключается в следующем: Пользователь переходит по ссылке, ведущей на веб-сайт приложения. Контроллер обрабатывает это действие и открывает веб-форму (представление) авторизации. Затем происходит процесс авторизации пользователя (также при взаимодействии контроллера, представления и модели). Затем, в зависимости от типа пользователя (обычный пользователь или администратор), открывается доступ к различным функциям приложения (добавление, удаление, редактирование, поиск). Все эти процессы связаны с базой данных пенсионных выплат ПФРФ, которая также расположена на сервере, и производятся через контроллер. Собственно, веб-приложение выступает переходным звеном во взаимодействии сотрудников ПФР (пользователей приложения) базы данных ПФР, используя разделение задач по принципу MVC. Процесс работы с приложением показан ниже на диаграмме деятельности (рисунок 1):

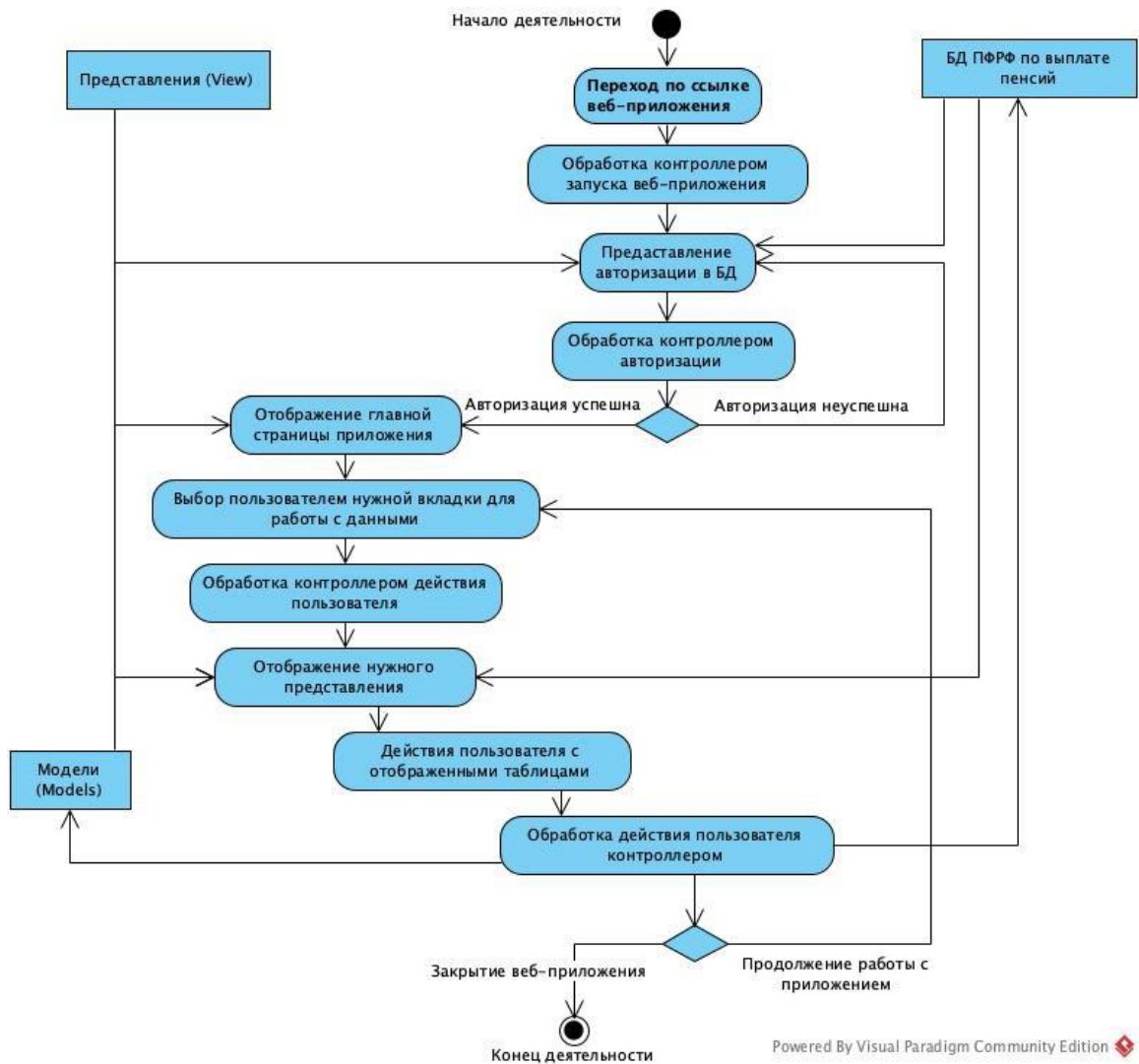


Рисунок 1 – Диаграмма деятельности работы веб-приложения

Рассмотрим процесс удаления сведений о выплатах пенсии из БД ПФР используя наше веб-приложение. После авторизации мы переходим во вкладку «Пенсии» и видим таблицу выплат.

Value	Date	Region	
12000	24.03.2021	Прокопьевск	Изменить Подробнее Удалить
14450	17.03.2021	Прокопьевск	Изменить Подробнее Удалить
17200	29.03.2021	Кемерово	Изменить Подробнее Удалить

Рисунок 2 – Вкладка «Пенсии» нашего веб-приложения.

Выбираем нужную нам выплату (Например, последнюю от 29.03.2021) и нажимаем «Удалить».

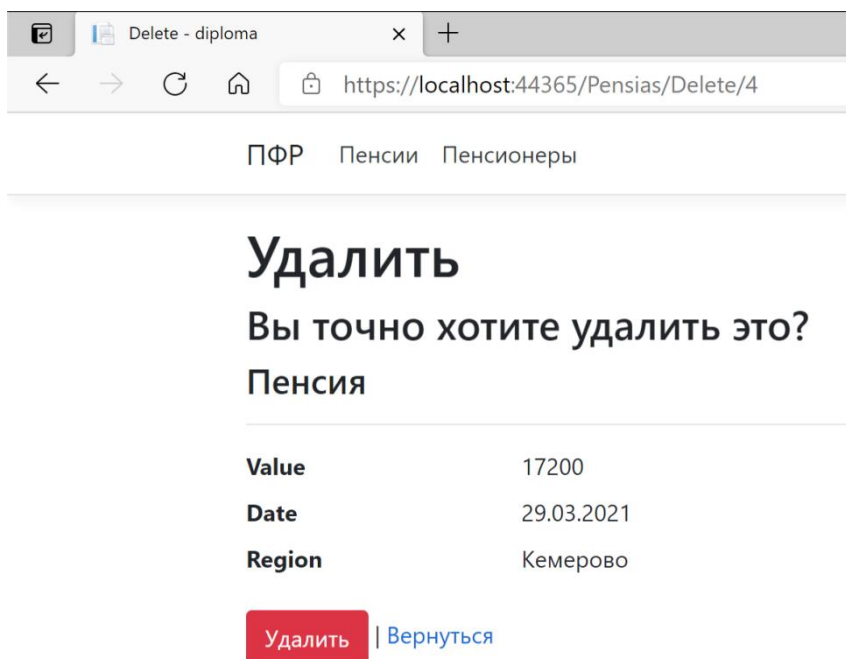


Рисунок 3 – удаление данных из таблицы «Пенсии»

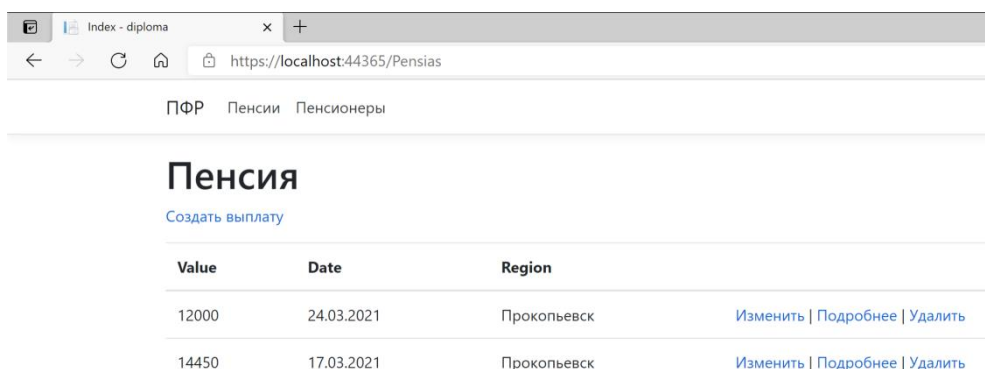


Рисунок 4 – Таблица «пенсии» после удаления данных

Это все действия, которые требуются от пользователя для удаления данных о выплате. В то же время, внутри веб-программы происходит взаимодействие представления (view) с контроллером (controller). Последний, в свою очередь, взаимодействует с моделью (model) и осуществляет взаимодействие с базой данных, из которой удаляется соответствующая запись в соответствующей таблице. Общий процесс лучше всего будет проиллюстрировать на диаграмме последовательности (рисунок 5):

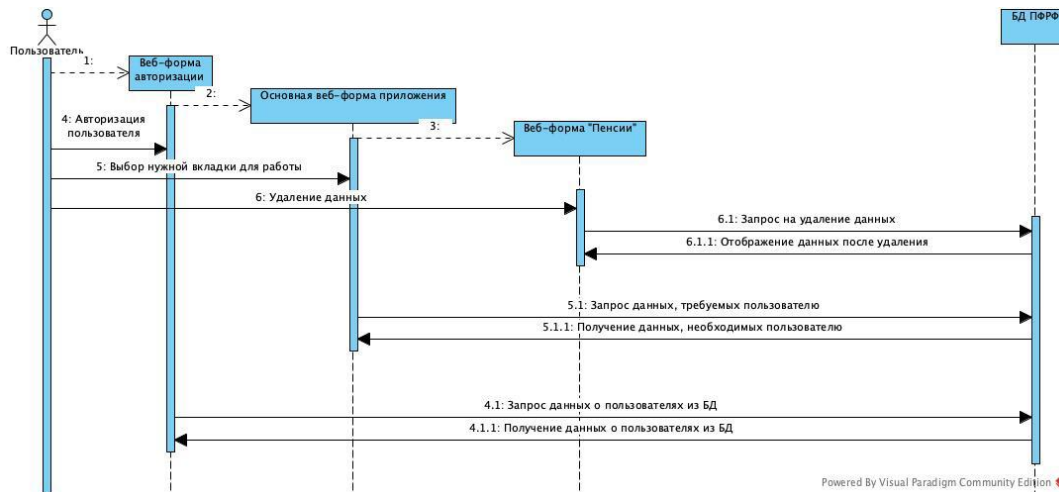


Рисунок 5 – диаграмма последовательности удаления данных из таблицы «Пенсии»

Таким образом, используя ASP.Net MVC Core, мы облегчаем разработку, поддержку и обновление нашего приложения. Помимо этого, мы можем разрабатывать приложение на разные операционные системы (благодаря MVC). А это обеспечивает больший охват при выборе ОС и удешевляет разработку (так как имеется возможность выбрать бесплатные ОС, например). Но при этом нужно учитывать повышенную нагрузку на сервер (на котором будет храниться веб-приложение) и на основе этого выбирать подходящее серверное оборудование (или улучшать текущее). Также, следует использовать усиленные методы защиты информации. В том числе непрограммные. Это подразумевает, например, обучение сотрудника мерам безопасности данных при работе из дома.

Список литературы:

1. Общие сведения ASP.Net Core MVC. [Электронный ресурс] URL: <https://docs.microsoft.com/ru-ru/aspnet/core/mvc/overview?view=aspnetcore-3.1> (дата обращения: 22.03.2021).
2. ASP.NET Core: Создание первого веб-API с использованием ASP.NET Core MVC и Visual Studio. [Электронный ресурс] URL: <https://habr.com/ru/company/microsoft/blog/312878/> (дата обращения: 21.03.2021).
3. Справочник по C#. [Электронный ресурс] URL: <https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/> (дата обращения: 20.03.2021).
4. Visual Paradigm User's Guide. [Электронный ресурс] – URL: <https://www.visual-paradigm.com/support/documents/vpuserguide.jsp> (дата обращения: 19.03.2021).

