

После решения задачи и отправки ответа в текстовый канал приложение сообщает об успешности его выполнения (рисунок 6). В случае отправки неверного ответа сообщение будет содержать соответствующую информацию с предложением пользователю попробовать снова отправить верный ответ. Также ответное сообщение содержит выпадающее меню с разделами тем, на которые после выбора можно будет выбрать желаемое задание.

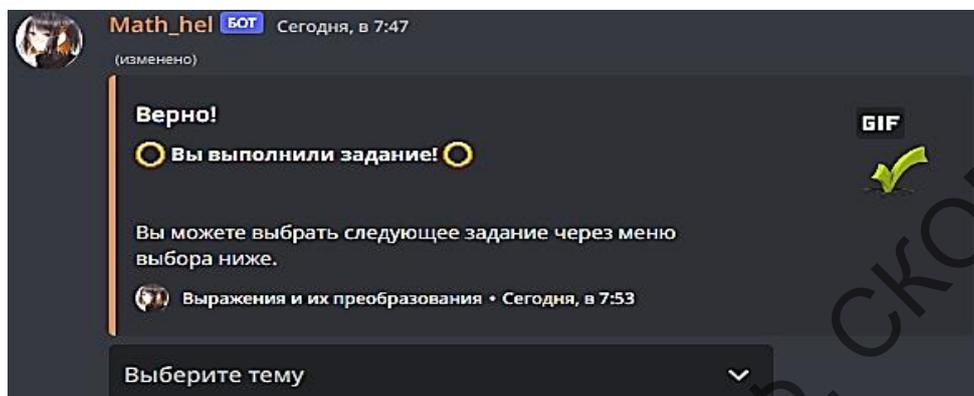


Рисунок 6 – Сообщение об успешном выполнении задания

Разработанное веб-приложение реализует основные функции дистанционного образовательного процесса. В дальнейшем планируется развивать данное веб-приложение, расширив его функционал.

Литература

1 Discordjs.guide Reactions [Электронный ресурс]. – Режим доступа: <https://discordjs.guide/popular-topics/reactions.html#reacting-to-messages>. – Дата доступа: 25.11.2022.

2 Discordjs.guide Updating from v13 to v14 [Электронный ресурс]. – Режим доступа: <https://discordjs.guide/additional-info/changes-in-v14.html#before-you-start>. – Дата доступа: 29.11.2022.

3 Discordjs.guide Select menus [Электронный ресурс]. – Режим доступа: <https://discordjs.guide/interactions/select-menus.html#building-and-sending-select-menus>. – Дата доступа: 25.05.2022.

УДК 004.4'2:004.77:656.11

Д. Г. Логунов

ОБРАБОТКА ИНФОРМАЦИОННЫХ ПОТОКОВ РАЙОННОГО ОТДЕЛЕНИЯ ГАИ С ИСПОЛЬЗОВАНИЕМ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ ПРОГРАММИРОВАНИЯ

В статье приводится описание программного инструментария, позволяющего организовать хранение, обработку и передачу служебной информации в процессе работы районного отделения ГАИ. Описаны современные технологии программирования, использованные при разработке приложения, такие как среда разработки базы данных, фреймворк для работы с базой данных, паттерн архитектуры приложения.

Введение. Информационные потоки данных, объем и скорость появления которых неуклонно возрастают с каждым днем, являются необходимой составляющей принятия решений сотрудниками ГАИ в ходе оценки текущих дорожных ситуаций и при прогнозировании различных вариантов их изменения. Большинство операций с данными не может быть выполнено вручную, и требуются программные средства, автоматизирующие обработку больших объёмов информации, позволяющие их структурировать и обеспечивающие оперативный доступ к ним уполномоченным сотрудникам Госавтоинспекции.

В статье описывается приложение, автоматизирующее получение, обработку, хранение и предоставление информации с использованием технологии WPF (Windows Presentation Foundation), фреймворка EntityFramework, паттерна MVVM, Dependency Injection и языка программирования C# [1]. Технология WPF является частью экосистемы платформы .NET и представляет собой подсистему для построения графических интерфейсов. Если при создании традиционных приложений на основе WinForms за отрисовку элементов управления и графики отвечали части ОС Windows, то приложения, разработанные с помощью WPF, основаны на DirectX. В этом состоит ключевая особенность рендеринга графики в WPF. Одной из важных особенностей является использование языка декларативной разметки интерфейса XAML, основанного на XML.

Описание базы данных приложения. Информация является ключевым элементом в деятельности сотрудников районного отделения ГАИ при выполнении следующих функций: регистрации автотранспортных средств; выполнении мер, повышающих безопасность дорожного движения; борьбе с угоном автотранспортных средств; поиске угнанных средств.

Вся информация, необходимая для работы приложения, была структурирована и согласована между собой. Для этого была изучена предметная область, выполнен её анализ, результатом которого стали основные сущности предметной области и связи между ними, которые послужили основой базы данных, предназначенной для хранения, обработки и фильтрации потоков служебной информации.

Средой разработки базы данных для приложения «Работа районного отделения ГАИ» была выбрана система управления базами данных MS SQL Server. Спроектированная база данных состоит из 9 таблиц (рисунок 1). Часть таблиц используется для хранения справочной информации, т. е. информации, которая используется часто, но редко изменяется. Это такие таблицы, как Автомобили, Водители, Виды нарушений, Сотрудники, Пользователи.

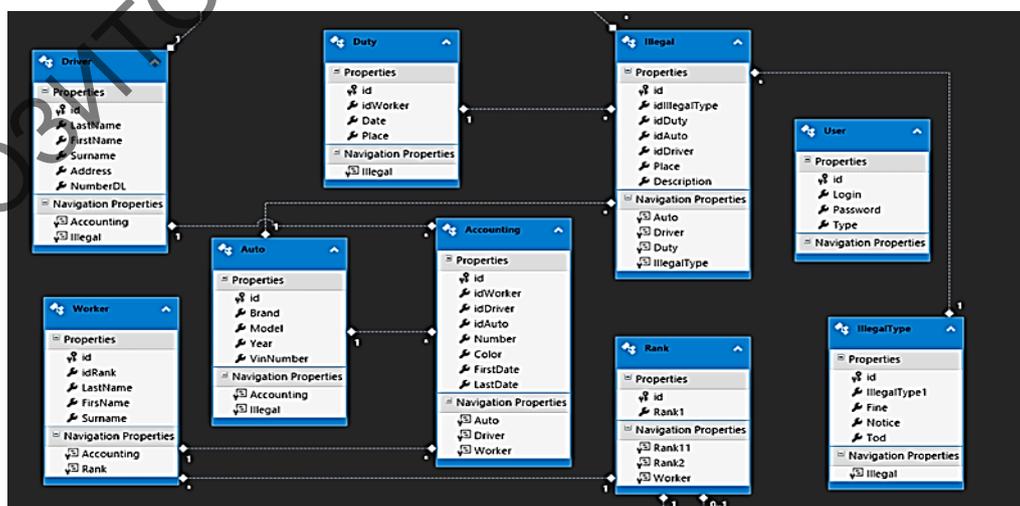


Рисунок 1 – База данных приложения

Данные таблиц Сотрудники, Дежурства позволяют хранить информацию как о сотрудниках районного отделения ГАИ, так и об их службе. Для каждого сотрудника в базе данных хранятся сведения обо всех его дежурствах, т. е. где и когда дежурил.

Часть таблиц предназначена для информации о постановке и снятии с учёта автомобилей граждан района, а также нарушениях правил дорожного движения водителями (Автомобили, Учет, Водители, Нарушения ПДД). Это позволит при использовании приложения в любой момент времени ответить на все вопросы, касающиеся регистрации автомобилей и их владельцев, а также узнать всю статистику по нарушениям правил дорожного движения и выявить злостных водителей-нарушителей. Дополнительно в базе данных будет храниться информация о дорожно-транспортных происшествиях на улицах района.

Структура приложения «Работа районного отделения ГАИ». Для разработанной базы данных была создана модель с использованием EntityFramework (рисунок 2). Данный фреймворк представляет собой объектно-ориентированную, легковесную и расширяемую технологию, разработанную компанией Microsoft для доступа к данным.

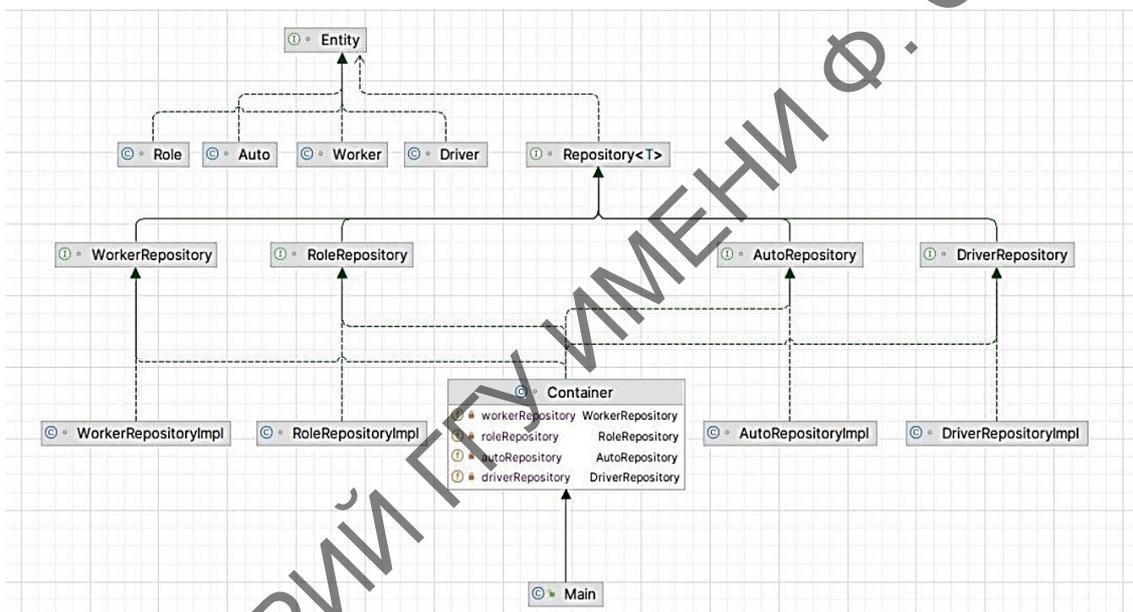


Рисунок 2 – Структура приложения «Работа районного отделения ГАИ»

Центральной концепцией данного фреймворка является использование сущностей или Entity. Любая сущность, как и любой объект из реального мира, обладает рядом свойств. Сущность определяет набор данных, связанных с определенным объектом, в частности, с разработанной базой данных для районного отделения ГАИ. Поэтому данная технология предполагает работу не с таблицами, а с объектами и их коллекциями.

При разработке приложения был использован паттерн MVVM (Model-View-ViewModel). Данный паттерн является архитектурным, то есть он задает общую архитектуру приложения. Также он позволяет отделить логику приложения от визуальной части (представления). Данный паттерн состоит из трех основных компонентов: модели (Model), модели представления (ViewModel) и представления (View). Модель описывает используемые в приложении данные. Модели могут содержать логику, непосредственно связанную с этими данными. В то же время модель не должна содержать никакой логики, связанной с отображением данных и взаимодействием с визуальными элементами управления. Представление определяет

визуальный интерфейс, через который пользователь взаимодействует с разработанным приложением. Представление в WPF – это код в xaml, который определяет интерфейс в виде кнопок, текстовых полей и прочих визуальных элементов.

Модель представления связывает модель и представление через механизм привязки данных. Если в модели изменяются значения свойств, то автоматически идет изменение отображаемых данных в представлении, хотя напрямую модель и представление не связаны. Также модель представления содержит логику по получению данных из модели, которые потом передаются в представление, определяет логику по обновлению данных в модели.

Логика взаимодействия с базой данных, взаимодействия с графическим интерфейсом была внедрена с помощью Dependency Injection.

Оформление интерфейса приложения. Приложение состоит из окна авторизации с разделением пользователей по ролям, главного меню и подпунктов. В главном меню предоставлен функционал, который доступен пользователю в зависимости от выданной роли (администратор, оператор, начальник). В главном меню, предусмотрены такие пункты меню как «Справочники», «ДТП», «Нарушения», «Дежурство», «Учёт», «Кадры». В данных пунктах меню приложения данные структурированы (рисунок 3). Например, в пункте меню «Дежурство» с помощью разработанной логики, организующей взаимодействие с базой данных, была выведена вся необходимая информация для работы с данным пунктом, т. е. предоставлены структурированные данные из таблиц «Дежурство», «Сотрудники».



Рисунок 3 – Интерфейс приложения

Заключение. В процессе проектирования приложения была разработана модель базы данных, спроектирована сама база данных, определена логика работы приложения, предоставляющего работникам удобный интерфейс для взаимодействия с базой данных и автоматизирующего основные функции районного отделения Госавтоинспекции. Разработанное программное обеспечение является универсальным и может быть использовано в любом районном отделении ГАИ.

Литература

1 Троелсен, Э. С# и платформа .NET. Библиотека программиста / Э. Троелсен. – Санкт-Петербург : Питер, 2004. – 796 с.