**Содержание**

[Введение 4](#_Toc160028944)

[1. Предварительное планирование и проверка осуществимости 6](#_Toc160028945)

[1.1 Требования к системе в целом 6](#_Toc160028946)

[1.2 Требования к функциям, выполняемым системой 8](#_Toc160028947)

[1.3 Требования к видам обеспечения 9](#_Toc160028948)

[1.4 Сравнение и анализ аналогов системы 10](#_Toc160028949)

[1.5 Инфологическая модель предметной области 13](#_Toc160028950)

[1.6 Прототипирование интерфейса 15](#_Toc160028951)

[2. Практическая часть 17](#_Toc160028952)

[2.1. Назначение и цели создания системы 17](#_Toc160028953)

[2.2 Характеристики объекта автоматизации 17](#_Toc160028954)

[2.3. Обзор инструментальных средств создания продукта 19](#_Toc160028955)

[2.4. Обоснование выбора средств реализации ИС 21](#_Toc160028956)

[2.5. Описание логики функционирования ПО 21](#_Toc160028957)

[2.6. Реализация интерфейса ПО 28](#_Toc160028958)

[2.7. Написание программного кода 33](#_Toc160028959)

[2.8. Защита информации 35](#_Toc160028960)

[Заключение 37](#_Toc160028961)

[Список использованных источников 38](#_Toc160028962)

[Приложения 39](#_Toc160028963)

[**Приложение А** 39](#_Toc160028964)

# Введение

В условиях постоянного развития технологий и повышенных требований к безопасности дорожного движения, автошколы сталкиваются с необходимостью совершенствования своих образовательных методов и технических решений. Развитие информационных технологий предоставляет уникальные возможности для оптимизации процессов обучения в автошколах, повышения эффективности преподавательской деятельности и обеспечения безопасности будущих водителей.

В современном обществе, где технологический прогресс становится неотъемлемой частью нашей повседневной жизни, образовательные учреждения, в том числе автошколы, не могут оставаться в стороне от высокотехнологичных трансформаций. С учетом постоянного увеличения числа учащихся и растущих требований к качеству обучения, создание эффективной информационной системы для автошколы становится неотложной задачей, который поможет автоматизировать ее процессы и упростит процесс получения информации, связанной с автошколой и расписанием.

**Актуальность темы** – Современные студенты ожидают доступа к образовательным ресурсам в интерактивной и цифровой форме. Информационные системы могут предоставить средства для учебного процесса, соответствующего ожиданиям молодого поколения.

**Цель** – разработка модуля для информационной системы для «Автошколы»

**Объект** – составление и просмотр расписания для курсантов автошколы

**Предмет** – модуль информационной системы для автошколы

**Для достижения цели, требуется выполнить следующие задачи:**

1. Провести предварительное планирование и проверку осуществимости.
2. Определить требования к программному продукту.
3. Разработать прототип приложения.
4. Реализовать приложение в выбранной среде IDE.
5. Разместить проект на веб-сервисе GitHub или аналогичном.

# Предварительное планирование и проверка осуществимости

# 1.1 Требования к системе в целом

Требования к структуре и функционированию системе

Система должна состоять из баз данных и приложения.

БД отвечает за хранение базы данных. Приложение организует работу приложения и обмен данными с базой данных. Приложение – основная часть системы – обрабатывает данные полученные от БД и выводит их в понятном для конечного пользователя виде, а также формирует запросы от пользователя к БД.

Требования к численности и квалификации персонала

Пользователями сайта являются потенциальные клиенты, сотрудники автошколы и курсанты. Пользователи используют систему через терминал, поэтому количество людей, желающих просмотреть или внести информацию через модуль информационной системы, неограниченно.

Обслуживающий персонал:

• администратор приложения и БД – 1 человек;

• модератор/работник автошколы – 1 человек.

Пользователи должны иметь минимальные навыки работы на компьютере.

Обслуживающий персонал должен иметь навыки работы на компьютере в среде Microsoft Windows (или другой операционной системе) и в глобальной сети Internet. Программист должен иметь опыт работы с вёрсткой сайта, MS SQL, Visual Studio 19 и выше, а также знать язык программирования C#.

Требования к надежности

Программное обеспечение системы должно функционировать и не приводить к зависанию системы при любых ошибочных действиях операторов в среде данного программного обеспечения.

Требования к эргономике и технической эстетике

Программное обеспечение и технические средства должны иметь эстетически привлекательный вид и быть удобными в использовании и должны быть интуитивно понятны для обычного пользователя.

Требования к эксплуатации и техническому обслуживанию

Для нормальной работы системы необходимо произвести ряд действий

на компьютере:

• установить лицензионную ОС;

• установить модуль информационной системы;

• запустить модуль.

• пройти авторизацию в программе.

на сервере базы данных:

• установить лицензионную ОС;

• установить SQL Server;

• Импортировать БД.

Требования к защите информации от несанкционированного доступа

Защита ИС и информации, с которой она работает, от несанкционированного доступа возлагается частично на операционную систему и специализированные программные и программно–аппаратные продукты, частично на систему аутентификации, реализованную в компонентах модуля ИС. Доступ к данным предоставляется только администратору и инструкторам.

Требования по сохранности информации при авариях

Сохранность данных должна быть обеспеченна:

• при сбоях в работе оборудования сети в любой момент времени;

• при сбоях электропитания или программного обеспечения не ведущих к физическому уничтожению или повреждению информации.

Требования к защите от влияния внешних воздействий

Автоматизированная система должна быть устойчивой к наличию и параллельной с ней работе на той же ЭВМ другого программного обеспечения.

Для уменьшения влияния перебоев с электропитанием рекомендуется питание «Сервера БД» производить через блок бесперебойного питания.

Требования к стандартизации и унификации

Автоматизированная система должна предоставлять пользователю привычный, общепринятый в среде Microsoft Windows интерфейс. Программная документация, поставляемая с программой, должна быть оформлена в соответствии со стандартом ЕСПД.

# 1.2 Требования к функциям, выполняемым системой

Функции подсистемы «Сервер базы данных»:

• получение запросов от «клиента БД», расположенного на SQL-сервере;

• проверка легальности выполнения запроса на основе идентификационной информации, вложенной в запрос;

• выполнение SQL-запросов к БД;

• передача результата SQL-запроса «клиенту БД».

Функции подсистемы «приложение»:

• передача пользователю следующей информации:

• информация о занятиях;

• контактная информация о преподавателях автошколы;

• информация о самой автошколе;

• изменение информации:

• изменение информации о занятиях;

• формирование и передача запросов серверу БД;

• получение от сервера БД результатов запросов.

Функции подсистемы «Клиент базы данных»:

• получение запросов приложения;

• получение от сервера БД результатов запросов.

# 1.3 Требования к видам обеспечения

Требования к математическому обеспечению

В основе разрабатываемого ПО должна лежать реляционная модель базы данных. Все таблицы должны находиться в третьей или четвертой нормальной форме. Связи между таблицами должны быть вида: «один к одному» или «один ко многим». При проектировании используется метод «сущность – связь».

Требования к информационному обеспечению

Данные в системе должны быть организованны в виде связанных таблиц. Хранение и доступ к данным осуществляется с применением реляционные систем управления базами данных. Информационный обмен между компонентами системы осуществляется по технологии клиент-сервер. Желательно обеспечить возможность резервного копирования и восстановления данных.

Требования к лингвистическому обеспечению

В качестве языка программирования используются – C#, XML и T-SQL, взаимодействие пользователя и программного обеспечения осуществляется оконный интерфейс модуля информационной системы.

Требования к программному обеспечению

Система должна иметь следующее программное обеспечение: ОС Windows 10 или выше, SQL Server.

Требования к техническому обеспечению

Для работоспособности сайта необходимы следующие программные и технические средства:

• клавиатура;

• манипулятор мышь;

• процессор Intel Core i3 и выше;

• оперативная память 4GB и больше;

• видеокарта 2 GB и монитор, поддерживающего разрешение 1024х768 точек и больше при глубине цвета 16 бит и более;

• дисковое пространство 16GB и больше.

# 1.4 Сравнение и анализ аналогов системы

|  |  |
| --- | --- |
| Наименование | Описание |
| [Информационная система учета и анализа деятельности автошколы «Академия вождения»](https://earchive.tpu.ru/handle/11683/38920) | В результате разработана информационная система, реализующая основные функции: учет договоров об оказании образовательных услуг и их оплаты; учет посещаемости занятий; учет и анализ сдачи внутренних и внешних экзаменов; анализ оказания услуг автошколы. |
| Изображение | |
|  | |
| Наименование | Описание |
| [Информационная система для автошколы](https://kursovik.com/programming/100450370.html) | Процесс обучения курсантов в автошколе |
| Изображение | |
|  | |
| Наименование | Описание |
| [Автоматизация организационных процессов в автошколе](https://core.ac.uk/download/pdf/84933988.pdf) | своевременное обеспечение участников процесса, надлежащей информацией касаемо: расписания занятий, контактных данных самих участников и актуальных новостей; контроль и регулирование учебного процесса курсантов и работы инструкторов со стороны начальства автошколы; упрощение записи на занятия; привлечение людей к обучению (посредством функции рекламы обучающимися, в социальных сетях). |
| Изображение | |
|  | |
| Наименование | Описание |
| [Информационная система автошколы](https://elib.pnzgu.ru/files/eb/doc/5VcM8gnpeyNG.pdf) | а) регистрация и авторизация пользователей; б) предоставление возможности изучения теории правил дорожного движения; в) предоставление возможность записи на практический урок вождения; г) предоставление возможность добавления водителей и данных о них; д) предоставление возможность просмотра статистики пройденных тестов по экзаменационным билетам правил дорожного движения |
| Изображение | |
|  | |

Сравнив эти проекты, можно сделать следующие выводы. Исходя из того, что все они созданы на основе сайта или мобильного приложения, интерфейс должен быть максимально упрощен для пользователя, так как моё приложение будет функционировать только на терминале. Следует уделить внимание на то, что ни одна из систем не предназначена для терминала, поэтому такая система будет иметь огромную уникальность и высокий интерес у пользователей. Если говорить о системах, расположенных на сайтах, можно отметить зависимость от Интернета. В отличии от них, система работающая без Интернета предотвратит большинство сбоев и ошибок при подключении и эксплуатации.

# 1.5 Инфологическая модель предметной области

Инфологическая модель представляет семантику (смысловое содержание предметной области в модели базы данных). Она должна включать в себя емкое описание предметной области, в котором можно было бы оценить корректность разработки проекта.

В инфологической модели содержится информация сущностях системы (объектах, имеющих важность в предметной области), атрибуты сущностей (свойства) и связями (отношениями) их с другими сущностями.

При разработке инфологической модели она должна соответствовать некоторым требованиям, таким как:

**•** Легкая расширяемость, которая позволяет ввести новые данные, не изменяя существующие;

**•** Возможность композиции и декомпозиции модели;

**•** Легкое восприятие различными классами пользователей (естественный язык при описании модели) и т.п.

Спроектированная ниже модель описывает последующие функции:

• Студенты: описывает информацию о студентах, их контактных данных, курсах, на которые они записаны, оплатах и прогрессе обучения;

• Инструкторы: содержит информацию об инструкторах, их квалификации, расписании занятий;

• Курсы: описывает информацию о доступных курсах, их стоимости, длительности и расписании занятий;

• Результаты экзамена: содержит информацию об курсанте и его оценке;

• Занятия: содержит в себе информацию о месте, дате, времени и продолжительности занятия;

• Пользователи: содержит в себе информацию о всех пользователях системы;

• Регистрация на курсы: данные для заполнения

Инфологическая модель для автошколы представляет собой описание данных и их взаимосвязей, которые используются в рамках

организации для управления ее деятельностью.

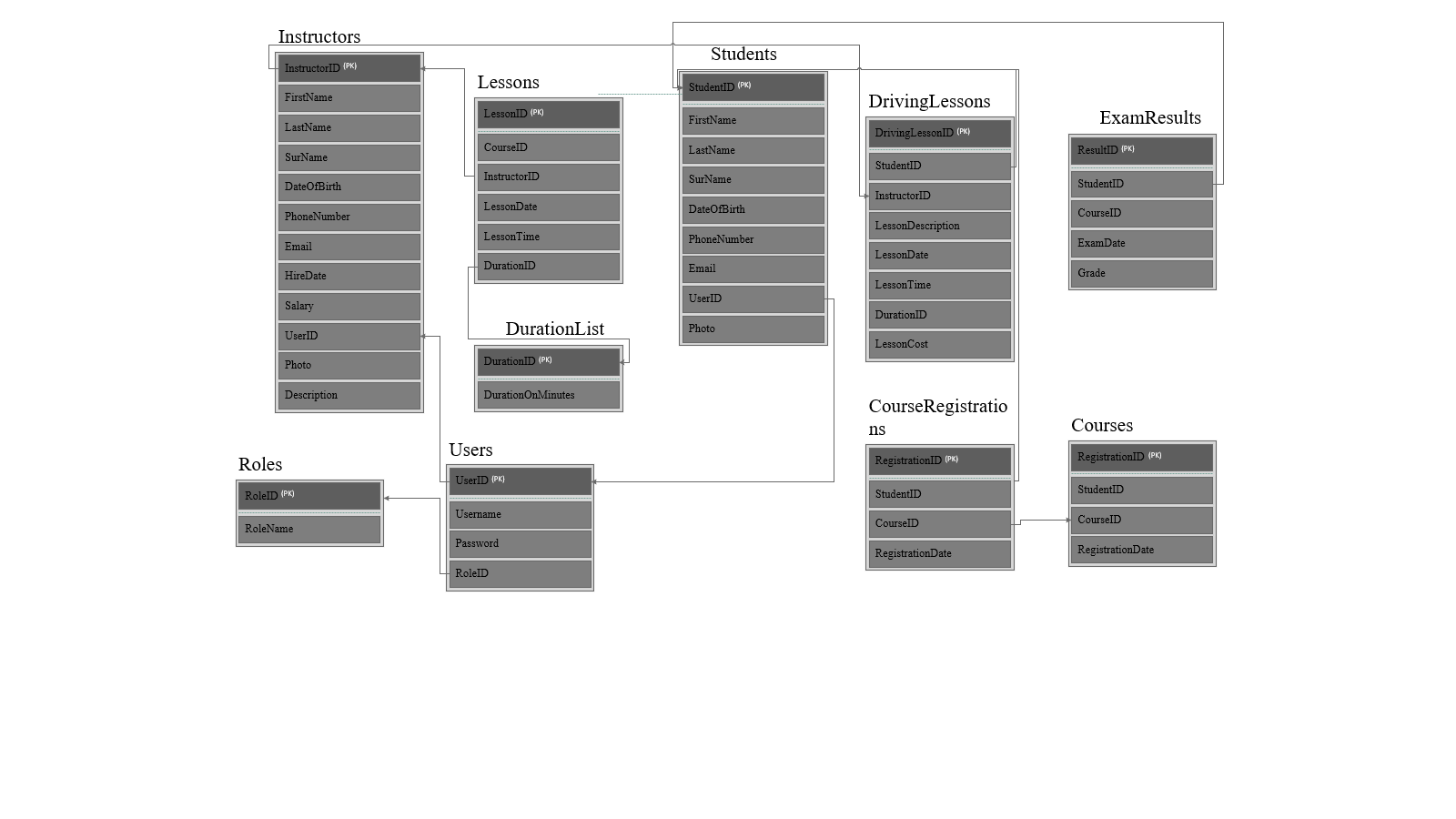


Рисунок 1.1. – «Инфологическая модель ПО»

# 1.6 Прототипирование интерфейса

Прототипирование является важным этапом в разработке веб-компонента, так как позволяет проверить идеи и концепции до того, как будет написан код. Это экономит время и ресурсы, так как можно быстро определить, что работает и что не работает, и внести необходимые изменения.

Прототип главной страницы имеет информацию о расписании, оплате, инструкторах и позволяет авторизоваться. На нем имеется 4 раздела, нажимая по ним пользователь будет переходить на страницы с тематикой соответствующего раздела. Главная страница представлена на рисунке 1.2.



Рисунок 1.2. – «Прототип страницы Главная»

Прототип страницы раздела «Расписание» включает в себя еще два дополнительных раздела, которые по нажатию открывают соответствующие разделы с необходимой информацией для обучения. Прототип страницы раздела Расписание представлен на рисунке 1.3.

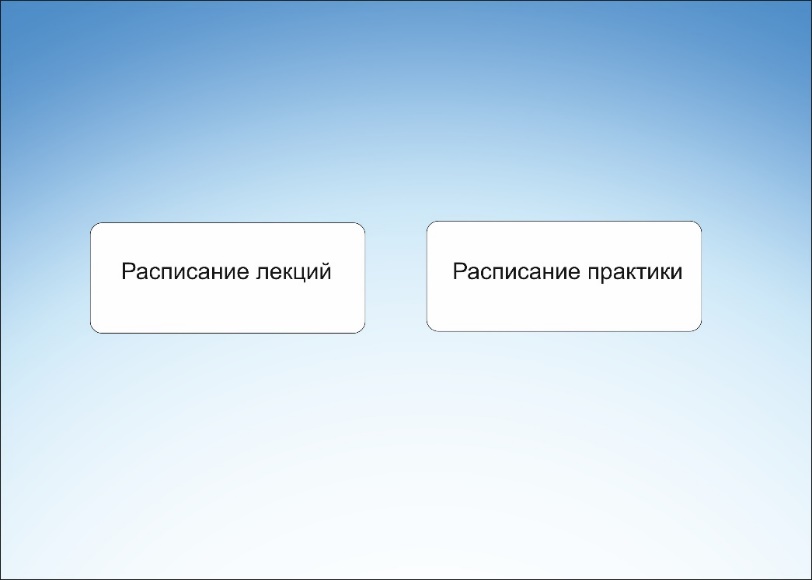


Рисунок 1.3. – «Прототип страницы Расписание»

Прототип страницы Авторизации включает в себя форму для ввода данных учетных записей пользователей, которых можно произвести вход в приложения по нажатию кнопки. Прототип страницы Авторизация представлен на рисунке 1.4.

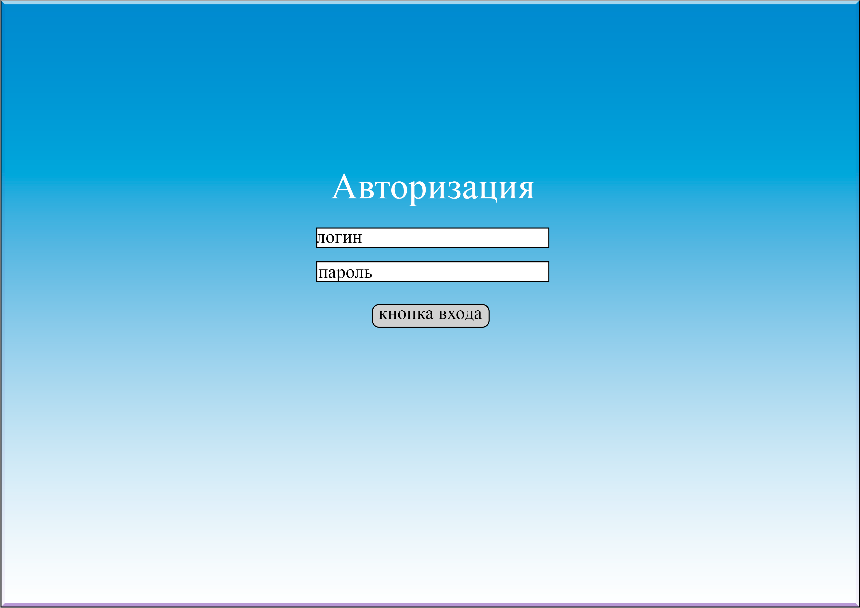


Рисунок 1.4. – «Прототип страницы Авторизация»

# 2. Практическая часть

# 2.1. Назначение и цели создания системы

Полное наименование программного обеспечения и его условное обозначение

Полное наименование – "Разработка модуля информационной системы для терминала автошколы «Дорожный ритм»”.

Назначение системы

Назначения создания модуля информационной системы для автошколы заключается в упрощении и автоматизации работы, что позволит повысить эффективность бизнес-процессов, снизить затраты на управление, повысить качество и скорость обслуживания клиентов, а также увеличить и сэкономить время как сотрудников, так и обучающихся.

Цели создания системы

• автоматизация процесса представления информации;

• помощь инструкторам в составлении расписания теоретических и практические занятий;

• повышение уровня доступности к предоставленной информации.

• упрощение процесса получения информации.

# 2.2 Характеристики объекта автоматизации

Краткие сведения об объекте автоматизации

Объекты автоматизации:

• процесс получения информации о курсанте;

• процесс получения информации о теоретических занятиях;

• процесс получения информации о практических занятиях;

• процесс получения информации об изменении расписания обучения;

• процесс получения информации об инструкторах автошколы;

В процессе выдачи информации разделах должны отображаться следующие данные:

• наименование раздела;

• информация о практических занятиях;

• информация о теоретических занятиях.

В процессе выдачи информации об инструкторах должны отображаться следующие данные:

• Ф.И.О;

• описание личных качеств;

• адрес, телефон, e-mail;

• рабочий стаж.

В процессе выдачи информации о расписании должны отображаться следующие данные:

• Дата занятия;

• Время занятия;

• Ф.И.О. курсанта

• Ф.И.О. инструктора

• Продолжительность занятия

Сведения об условиях эксплуатации объекта автоматизации и характеристик окружающей среды

Условия эксплуатации объекта автоматизации (на стороне приложения):

• операционная система MS Windows 10 или выше;

Условия эксплуатации объекта автоматизации (на стороне сервера БД):

• операционная система MS Windows 10 или выше;

• SQL Server;

Условия эксплуатации объекта автоматизации (на стороне клиента):

• любая операционная система;

Физические условия эксплуатации и характеристики окружающей среды определяются соответствующими характеристиками окружающей среды используемого аппаратного обеспечения, указанными в документации на него.

# 2.3. Обзор инструментальных средств создания продукта

Для реализации проекта можно воспользоваться следующими доступными средствами:

* Visual Studio – представляет собой полностью интегрированную среду разработки. Она спроектирована таким образом, чтобы делать процесс написания кода, его отладки и компиляции в сборку для поставки конечным потребителям как можно более простым. Является наиболее функционально насыщенной IDE-средой, в которой можно разрабатывать приложения на C#. Данный продукт поставляется с конструкторами графических пользовательских интерфейсов, с возможностями работы с базами данных, с поддержкой фрагментов кода, с возможностями для просмотра всего проекта в целом, с просмотром свойств объектов.
* Standard Widget Toolkit – библиотека графических пользовательских элементов с открытым исходным кодом для разработки интерфейсов пользователя на языке Java. Предоставляет из себя переносимый API для разработчиков Java. Использует собственные библиотеки операционной системы через собственный интерфейс Java (Java Native Interface) для визуализации элементов графического интерфейса, аналогично программам, написанным с использованием собственного API операционной системы.
* PyCharm – это интегрированная среда разработки на языке Python, которая была разработана международной компанией JetBrains (да, и снова эти ребята). Эта IDE распространяется под несколькими лицензиями, в том числе как Community Edition, где чуть урезан функционал. Сами разработчики характеризуют свой продукт как «самую интеллектуальную Python IDE с полным набором средств для эффективной разработки на языке Python», но она не ограничивается на Python. PyCharm поддерживает Google App Engine; IronPython, Jython, Cython, PyPy wxPython, PyQt, PyGTK и др.
* Qt Creator – кроссплатформенная свободная IDE для разработки на С, С++ и QML с помощью фреймворка Qt. Разработана Trolltech для работы с фреймворком Qt. Включает в себя графический интерфейс отладчика и визуальные средства разработки интерфейса как с использованием QtWidgets, так и QML. Зачастую используется при написания программного обеспечения для роботов, автомобилей и кораблей, которые используются на производствах, в перевозке грузов и других похожих отраслях. Qt Creator позволяет добиться высокой производительности при прототипировании интерфейса в приложениях любой сложности.
* CLion – продукт JetBrains. Многофункциональная IDE. В которой поддерживается разработка кода на C и C++, а также web-разработка на HTML(5), CSS, JavaScript, XML. А также на некоторых других языках доступнен в виде плагинов (например, Lua). Основным преимуществом является статический и динамический анализ с безопасный рефакторингом кода. Позволяет кастомизировать редактор так, как удобно пользователю.
* IntelliJ IDEA – ещё один продукт от JetBrains. Комплекс программных средств, который используется для написания, исполнения, отладки и оптимизации кода для многих языков программирования, в частности Java, JavaScript, Python и других языков программирования от компании JetBrains. Отличается обширным набором инструментов для рефакторинга (перепроектирования) и оптимизации кода. Интеграция с серверами приложений, включая Tomcat, TomEE, GlassFish и другие. Имеет набор инструментов для работы с базами данных, SQL файлами и инструменты для запуска тестов и анализа покрытия кода, включая поддержку всех популярных фреймворков для тестирования.

# 2.4. Обоснование выбора средств реализации ИС

Я решила, что буду писать данный курсовой проект в интегрированной среде разработки (IDE): Visual Studio. Она предоставляет мощную интегрированную среду разработки, которая объединяет в себе редактор кода, отладчик, компиляторы, а также другие инструменты для удобной работы над проектами любого размера.

Поддержка различных языков программирования: Visual Studio обладает широкой поддержкой различных языков программирования, таких как C++, C#, Visual Basic, F#, Python и другие. Это делает ее универсальным инструментом для разработчиков, работающих с разными технологиями.

Отличный отладчик: Visual Studio предлагает мощные средства отладки, позволяющие легко находить и исправлять ошибки в коде. Это значительно упрощает процесс разработки и повышает продуктивность разработчика.

Множество расширений и инструментов: Visual Studio обладает богатой экосистемой плагинов и дополнений, которые позволяют настраивать среду разработки под свои задачи и предпочтения, добавлять новые возможности и повышать эффективность работы.

Поддержка платформ: Visual Studio поддерживает разработку приложений для различных платформ, включая Windows, Linux, Android, iOS и многие другие. Это делает ее универсальным инструментом для разработки мобильных, веб- и десктоп-приложений.

Таким образом, Visual Studio является популярным выбором среди разработчиков благодаря своей функциональности, удобству использования и широким возможностям для работы с различными технологиями.

# 2.5. Описание логики функционирования ПО

При разработке модуля информационной системы станции технического обслуживания была создала база данных “AutoSchoolTerminal”, в которой хранится вся необходимая информация для корректной работы модуля информационной системы. Это информация о расписании, инструкторах, о списках учебных групп, их учетных данных. Также спроектирована схема базы данных, которая представлена на рисунке 2.1.

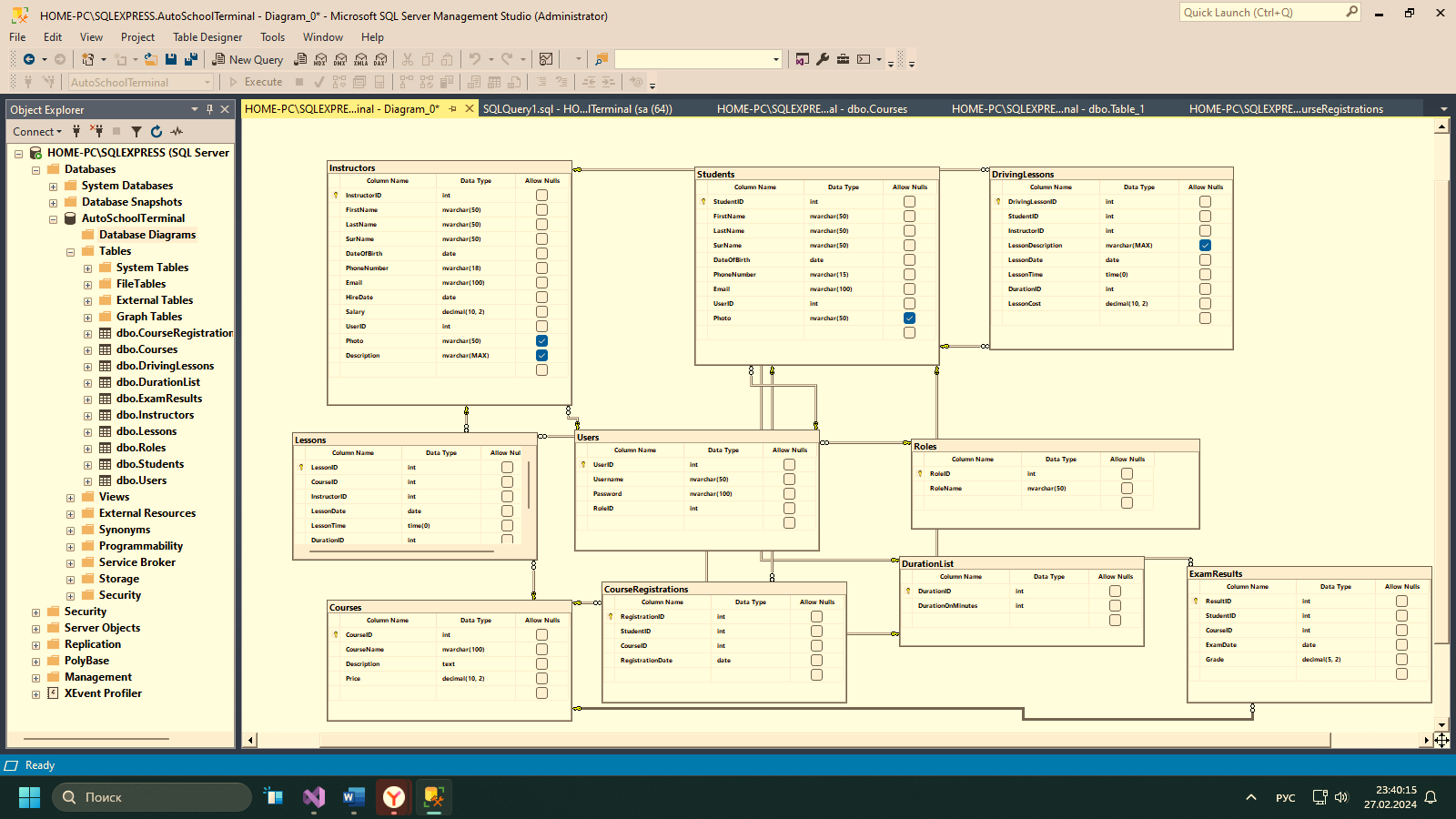


Рисунок 2.1. – “Схема базы данных AutoSchoolTerminal”

Таблица «CourseRegistrations» - Информация о регистрации на курс. Данные таблицы изображены на рисунке 2.2.

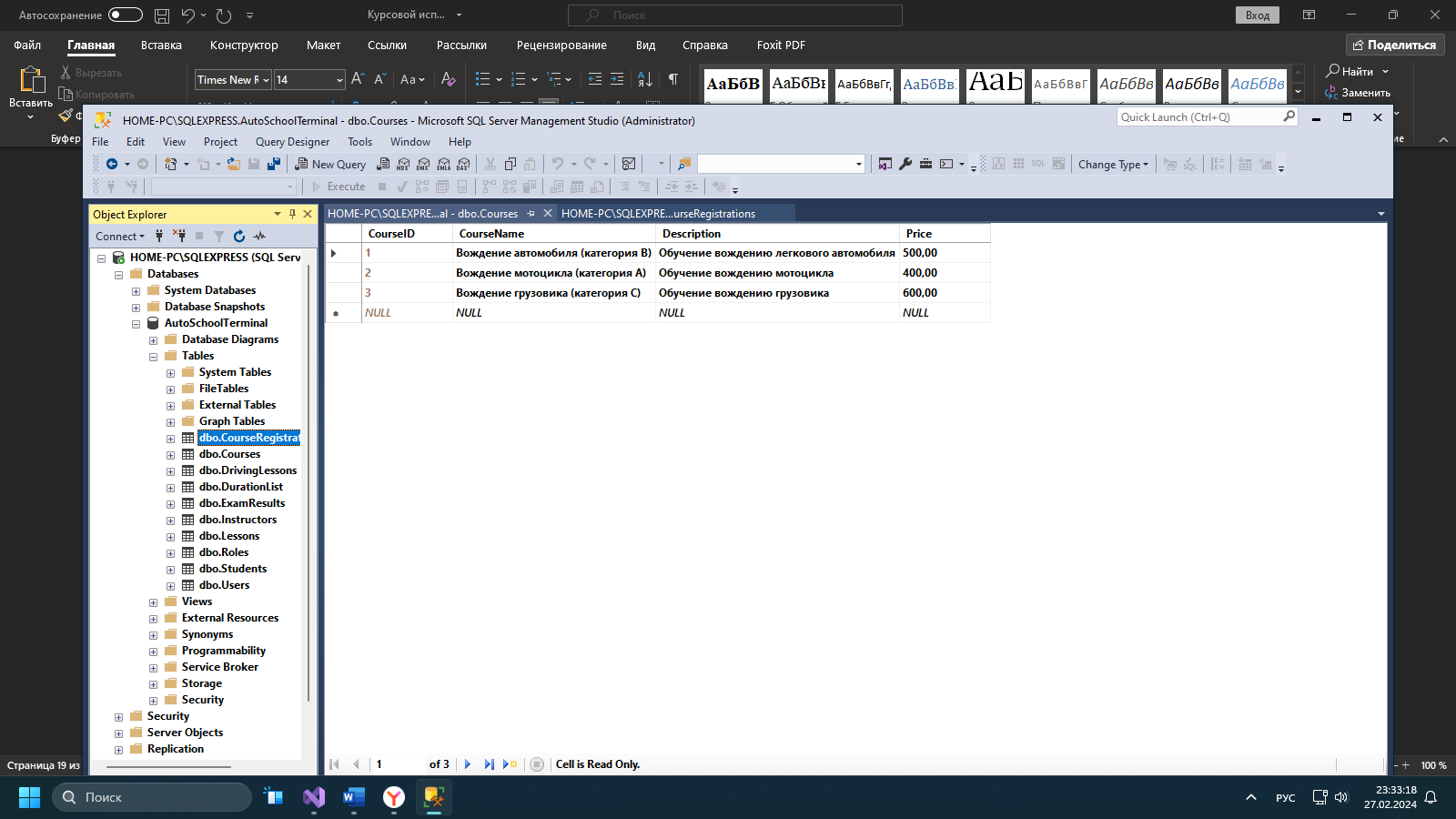


Рисунок 2.2. – «Данные таблицы CourseRegistrations»

Таблица 2.1. – Описание структуры CourseRegistrations

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| RegistrationID | Int | Id регистрации |
| StudentID | Int | Id студента |
| CourseID | Int | Id курса |
| RegistrationDate | Date | Дата регистрации |

Таблица «DrivingLessons» - Информация о занятиях по вождению. Данные таблицы изображены на рисунке 2.3.

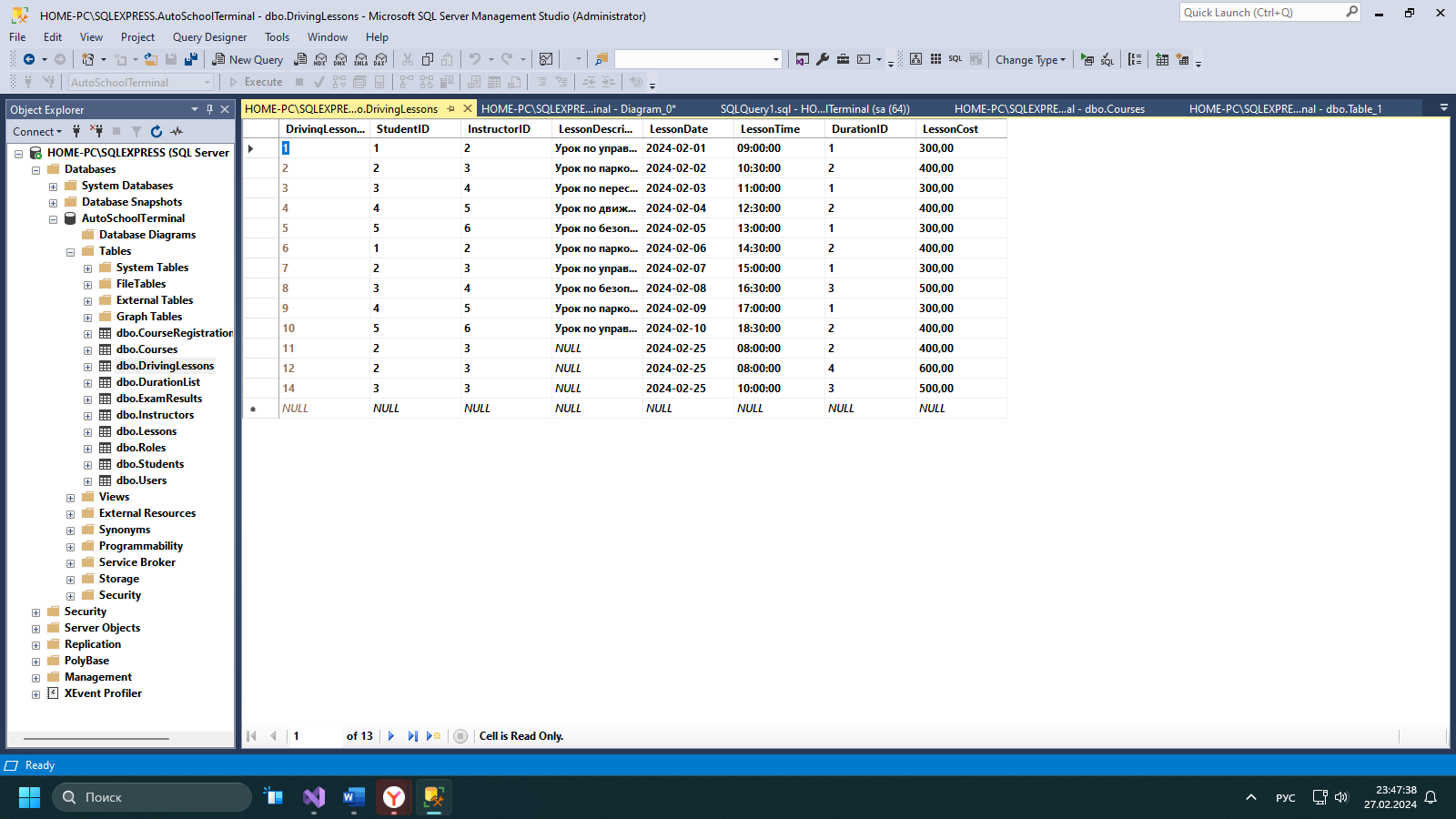


Рисунок 2.3. – «Данные таблицы DrivingLessons»

Таблица 2.2. – Описание структуры DrivingLessons

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| DrivingLessonID | int | Id занятия по вождению |
| StudentID | int | Id студента |
| InstructorID | int | Id инструктора |
| LessonDescription | nvarchar(MAX) | Описание занятия |
| LessonDate | date | Дата занятия |
| LessonTime | time(0) | Время занятия |
| DurationID | int | Продолжительность |
| LessonCost | decimal(10, 2) | Стоимость занятия |

Таблица «DurationList» - Информация о продолжительности. Данные таблицы изображены на рисунке 2.4.

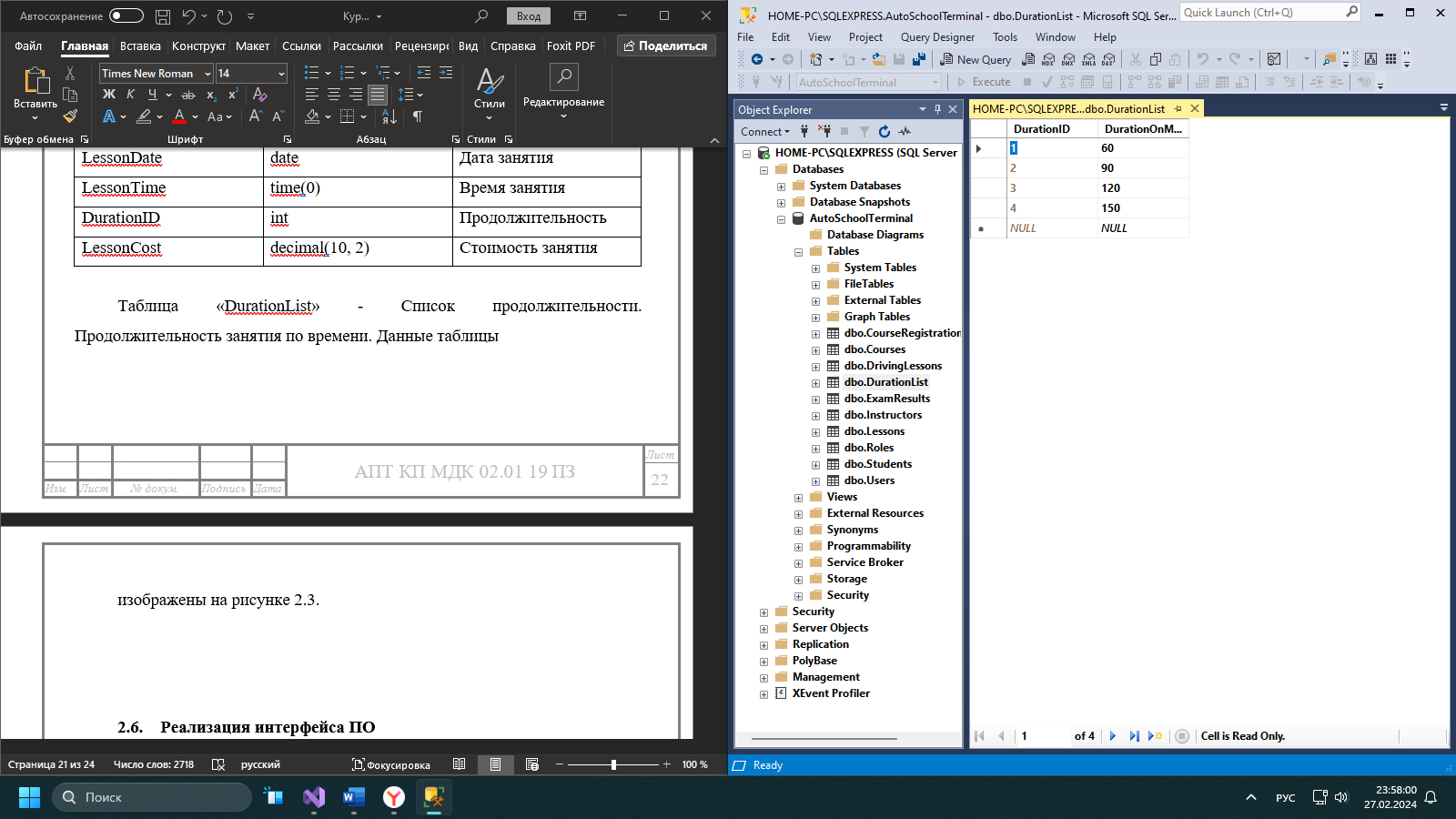


Рисунок 2.4. – «Данные таблицы DurationList»

Таблица 2.3. Описание структуры DurationList

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| DurationID | Int | Id продолжительности |
| DurationOnMinutes | int | Продолжительность в минутах |

Таблица «ExamResults» - Информация об результатах экзамена. Данные таблицы изображены на рисунке 2.5.

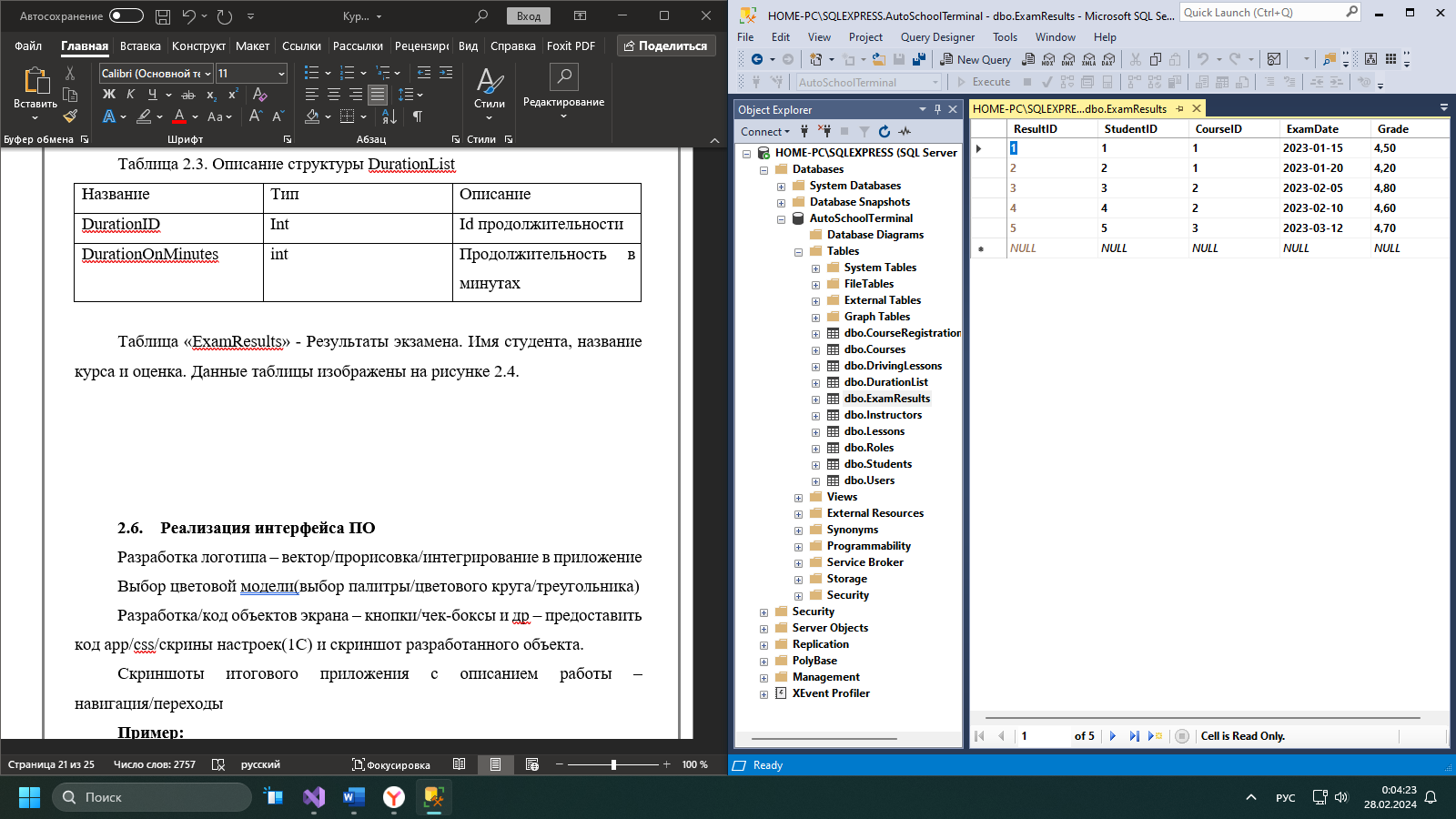


Рисунок 2.5. – «Данные таблицы ExamResults»

Таблица 2.4. Описание структуры ExamResults

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| ResultID | int | Id результата |
| StudentID | int | Id студента |
| CourseID | int | Id курса |
| ExamDate | date | Дата экзамена |
| Grade | decimal(5, 2) | Оценка |

Таблица «Instructors» - Информация об инструкторах. Данные таблицы изображены на рисунке 2.6.

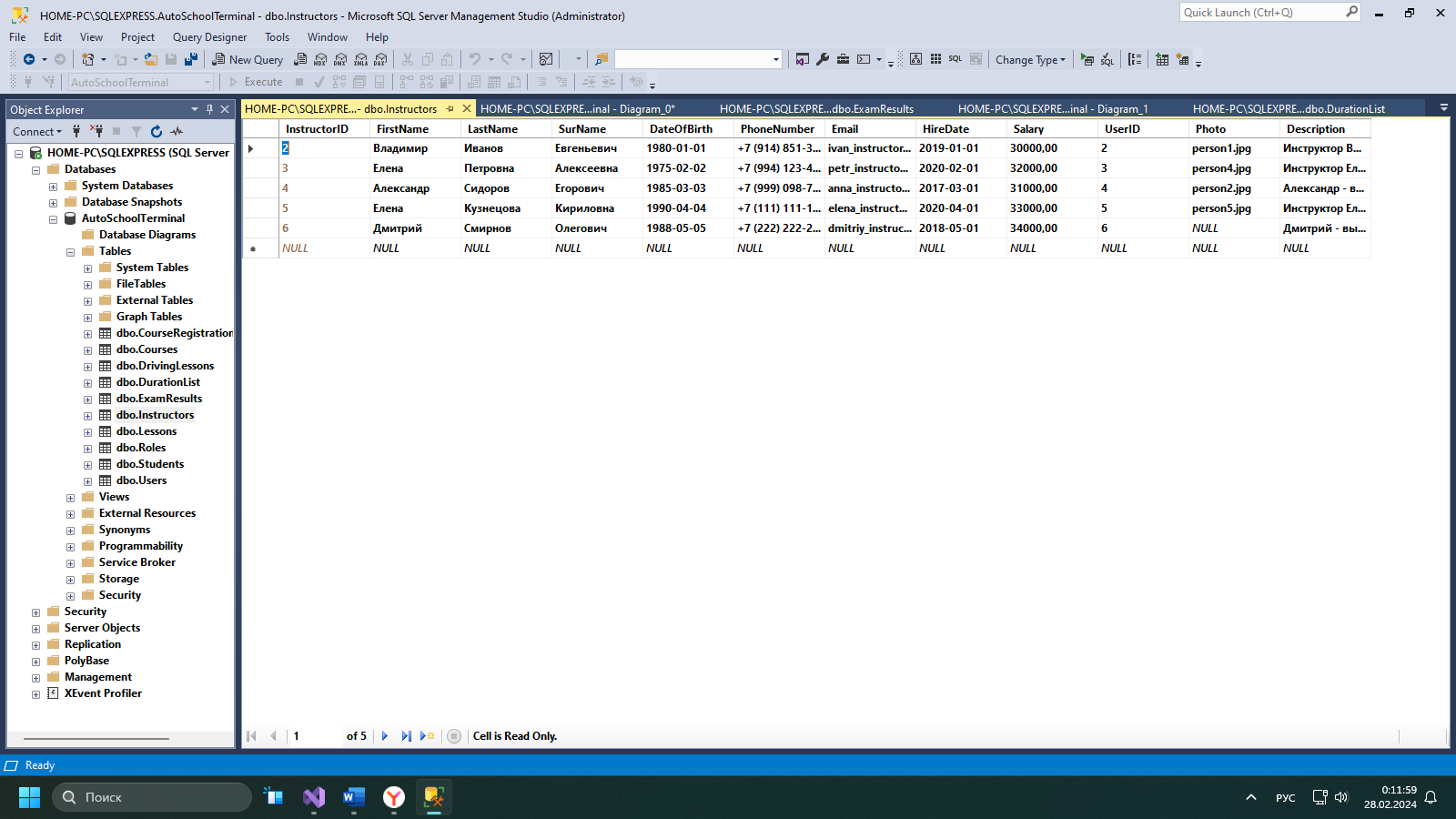


Рисунок 2.6. – «Данные таблицы Instructors»

Таблица 2.5. Описание структуры Instructors

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| InstructorID | int | Id инструктора |
| FirstName | nvarchar(50) | Фамилия инструктора |
| LastName | nvarchar(50) | Имя инструктора |
| SurName | nvarchar(50) | Отчество инструктора |
| DateOfBirth | date | Дата рождения инструктора |
| PhoneNumber | nvarchar(18) | Номер телефона |
| Email | nvarchar(100) | Электронная почта |
| HireDate | date | Дата приема на работу |
| Salary | decimal(10, 2) | Зарплата |
| UserID | int | Id пользователя |
| Photo | nvarchar(50) | Фото инструктора |
| Description | nvarchar(MAX) | Описание |

Таблица «Lessons» - Информация о занятиях. Данные таблицы изображены на рисунке 2.7.

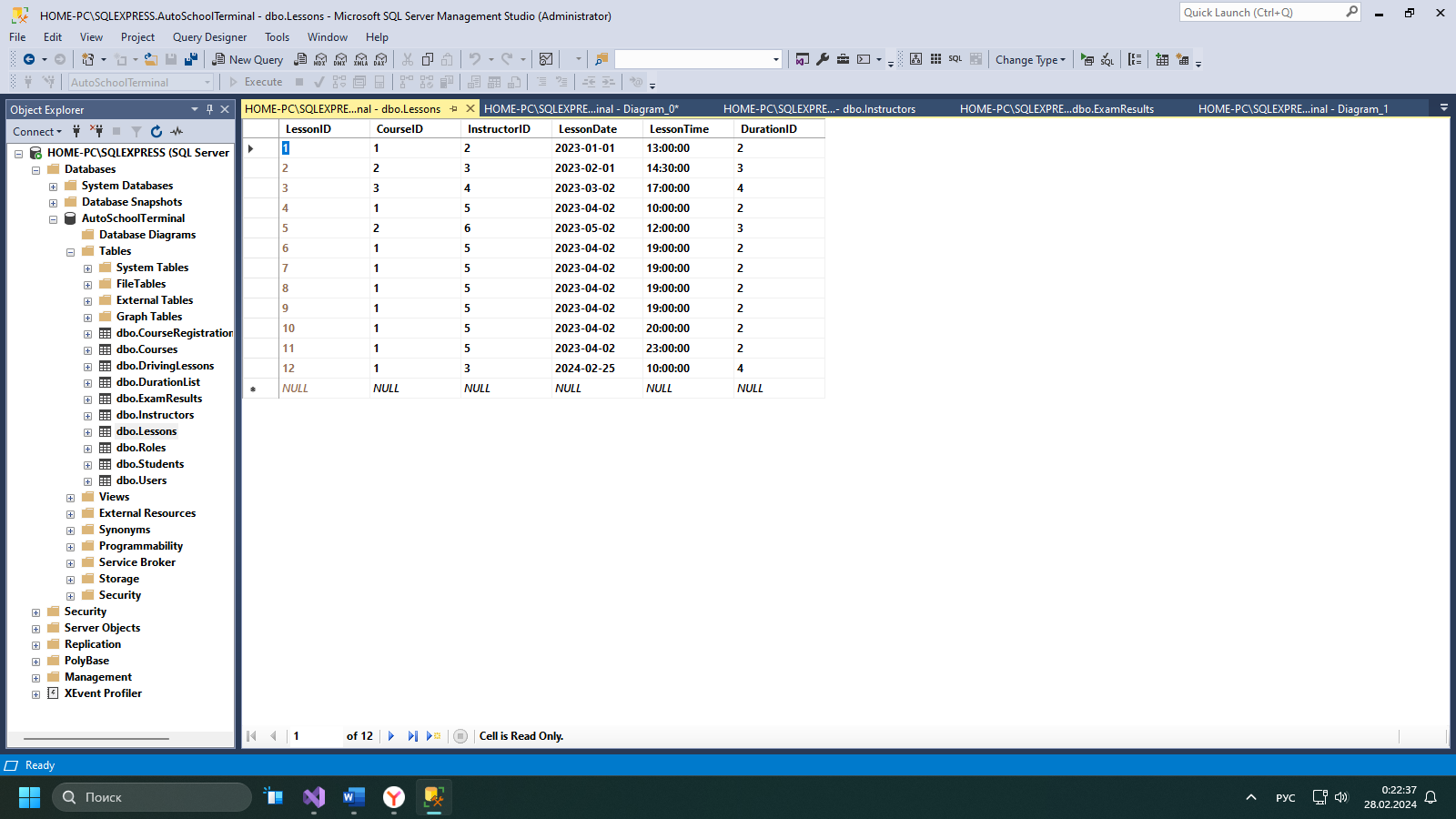


Рисунок 2.7. – «Данные таблицы Lessons»

Таблица 2.6. – Описание структуры Lessons

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| LessonID | int | Id занятия |
| CourseID | int | Id курса |
| InstructorID | int | Id инструктора |
| LessonDate | date | Дата занятия |
| LessonTime | time(0) | Время занятия |
| DurationID | int | Продолжительность занятия |

Таблица «Roles» - Информация о ролях. Данные таблицы изображены на рисунке 2.8.

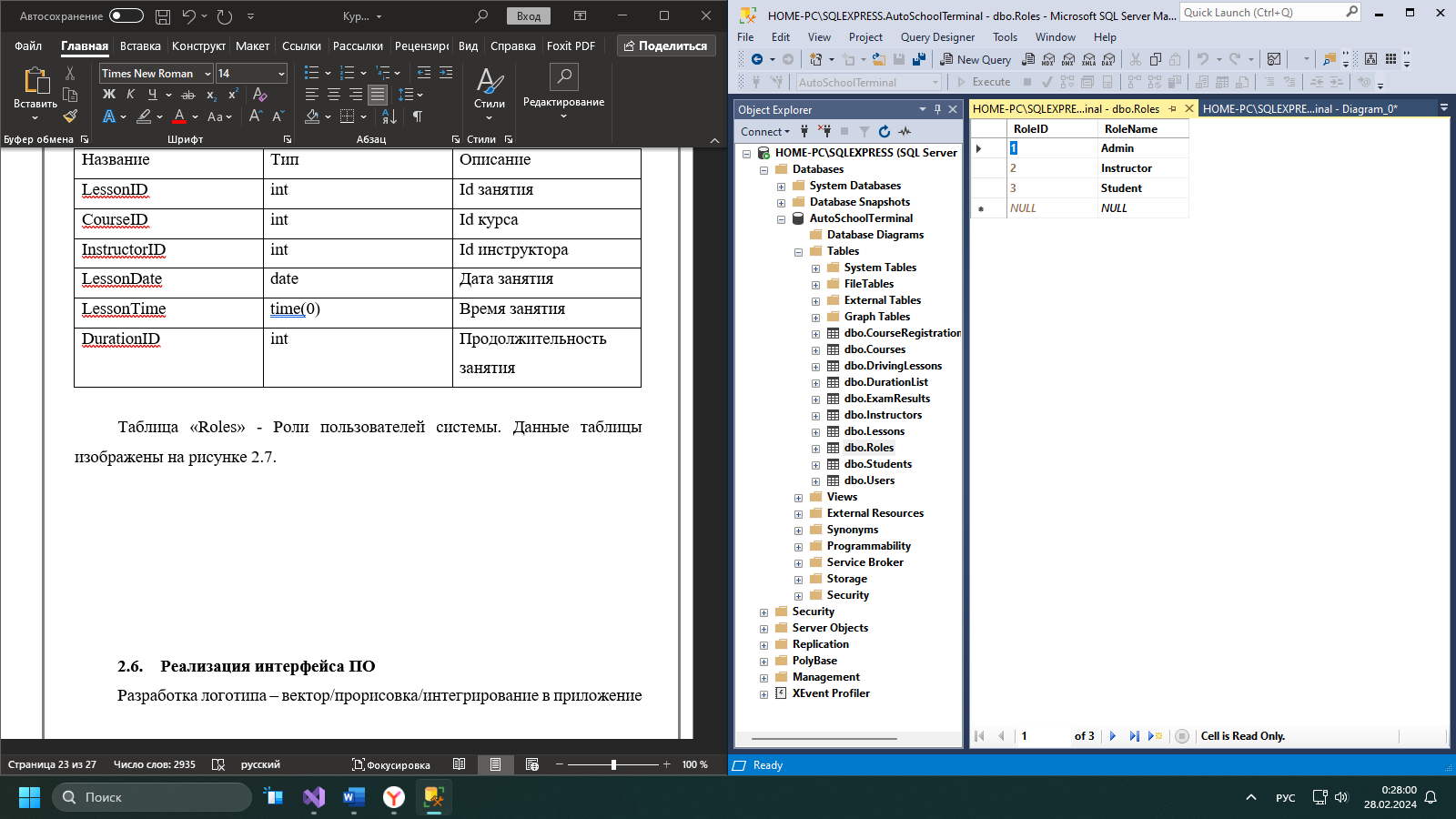


Рисунок 2.8. – «Данные таблицы Roles»

Таблица 2.7. Описание структуры Roles

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| RoleID | int | Id роли |
| RoleName | nvarchar(50) | Наименование роли |

Таблица «Students» - Информация о студентах. Данные таблицы изображены на рисунке 2.9.

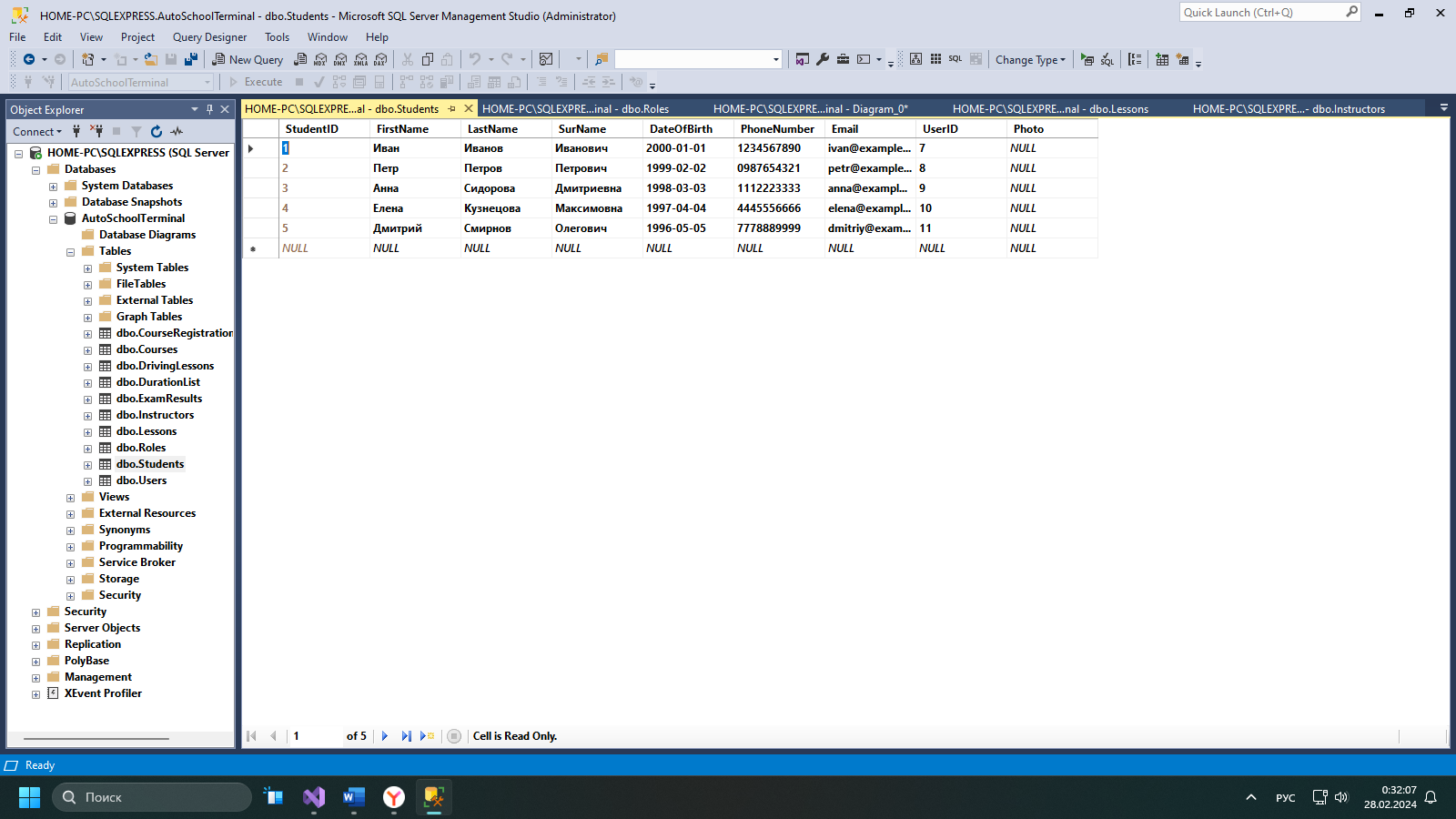


Рисунок 2.9. – «Данные таблицы Students»

Таблица 2.8. Описание структуры Students

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| StudentID | int | Id студента |
| FirstName | nvarchar(50) | Фамилия студента |
| LastName | nvarchar(50) | Имя студента |
| SurName | nvarchar(50) | Отчество студента |
| DateOfBirth | date | Дата рождения |
| PhoneNumber | nvarchar(15) | Номер телефона |
| Email | nvarchar(100) | Электронная почта |
| UserID | int | Id пользователя |
| Photo | nvarchar(50) | Фото студента |

Таблица «Users» - информация о пользователях. Данные таблицы изображены на рисунке 2.10.

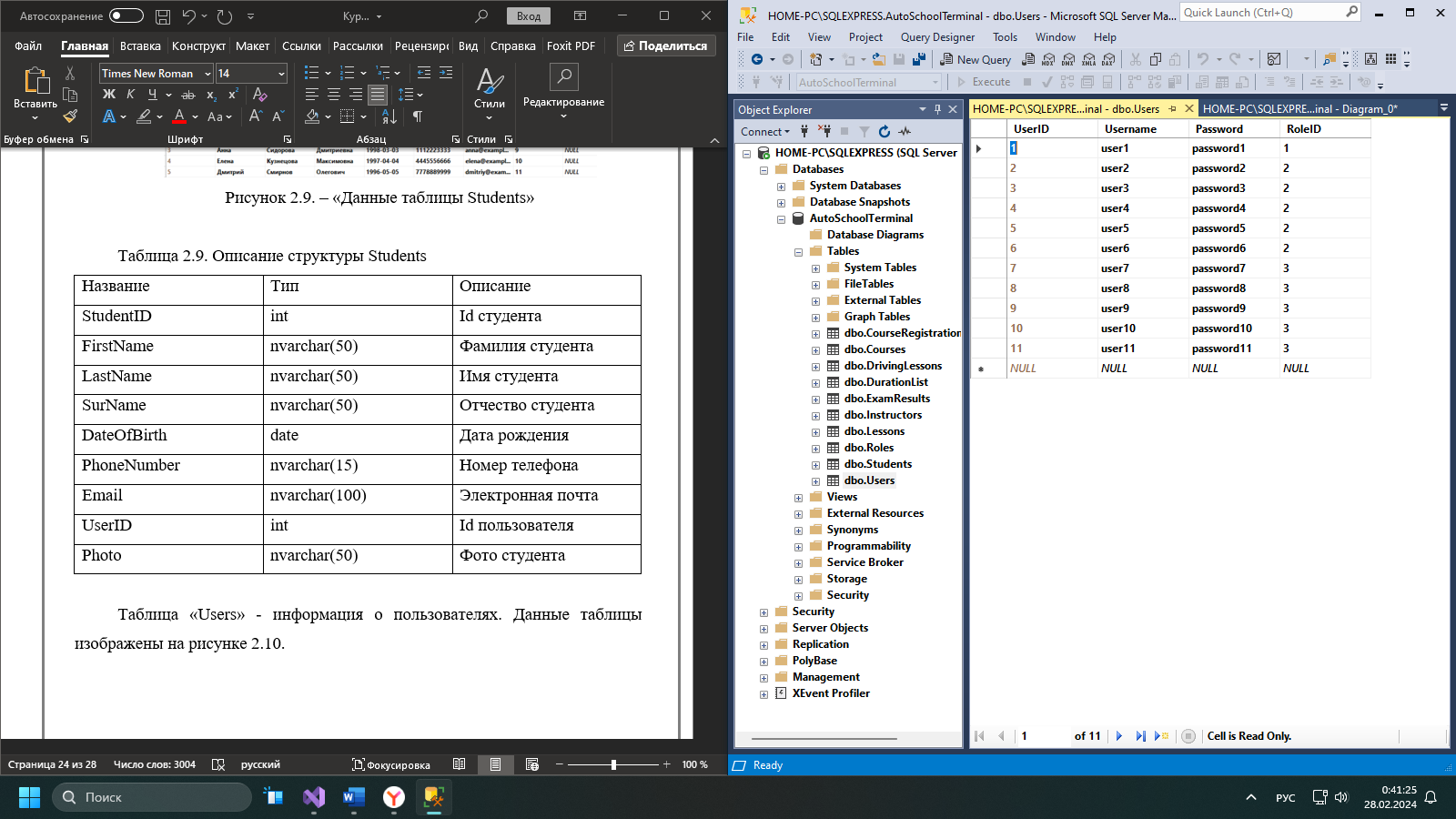


Рисунок 2.10. – «Данные таблицы Users»

Таблица 2.9. Описание структуры Users

|  |  |  |
| --- | --- | --- |
| Название | Тип | Описание |
| UserID | int | Id пользователя |
| Username | nvarchar(50) | Имя пользователя |
| Password | nvarchar(100) | Пароль |
| RoleID | int | Id роли |

# 2.6. Реализация интерфейса ПО

Хороший интерфейс программы делает ее использование удобным и эффективным для пользователя. Интерфейс программы является оконным и содержит несколько окон как для пользователя, так и для администратора.

В зависимости от критериев запуска приложения открываются разные окна, если приложение открыто от имени администратора, тогда открывается окно специалиста, где может взаимодействовать со всеми данными. В ином случае открывается окно пользователя или курсанта.

Для приложения я разработала индивидуальный логотип. Его особенности: простота и ясность, цветовая палитра, стиль, масштабируемость. Хороший логотип помогает узнаваемости бренда, создает положительное впечатление у потребителей и помогает выделиться на рынке. Логотип представлен на рисунке 2.11.



Рисунок 2.11. – «Логотип приложения»

Была выбрана цветовая палитра для компонента веб-системы. Палитра представлена в таблице 2.10.

Таблица 2.10. Цветовая модель приложения

|  |  |  |
| --- | --- | --- |
| Основной фон | Дополнительный фон | Акцентирование  внимания |
| RGB  #85CFF7  #FF340BFF | RGB  #85CFF7 | RGB  #FF340BFF |
| (133, 207, 247)  (255, 255,255) | (133, 207, 247) | (255, 255,255) |
|  |  |  |

При входе в приложение пользователя встречает стартовое окно с 4 кнопками, через которые он попадает в нужное ему окно и получает необходимую информацию для обучения, что показано на рисунке 2.12.



Рисунок 2.12. – «Главная страница»

Перейдя по кнопке «Личный кабинет» пользователь может зайти под 3 ролями системы. Первая – это администратор, он имеет возможность редактировать расписание, она показана на рисунке 2.13.

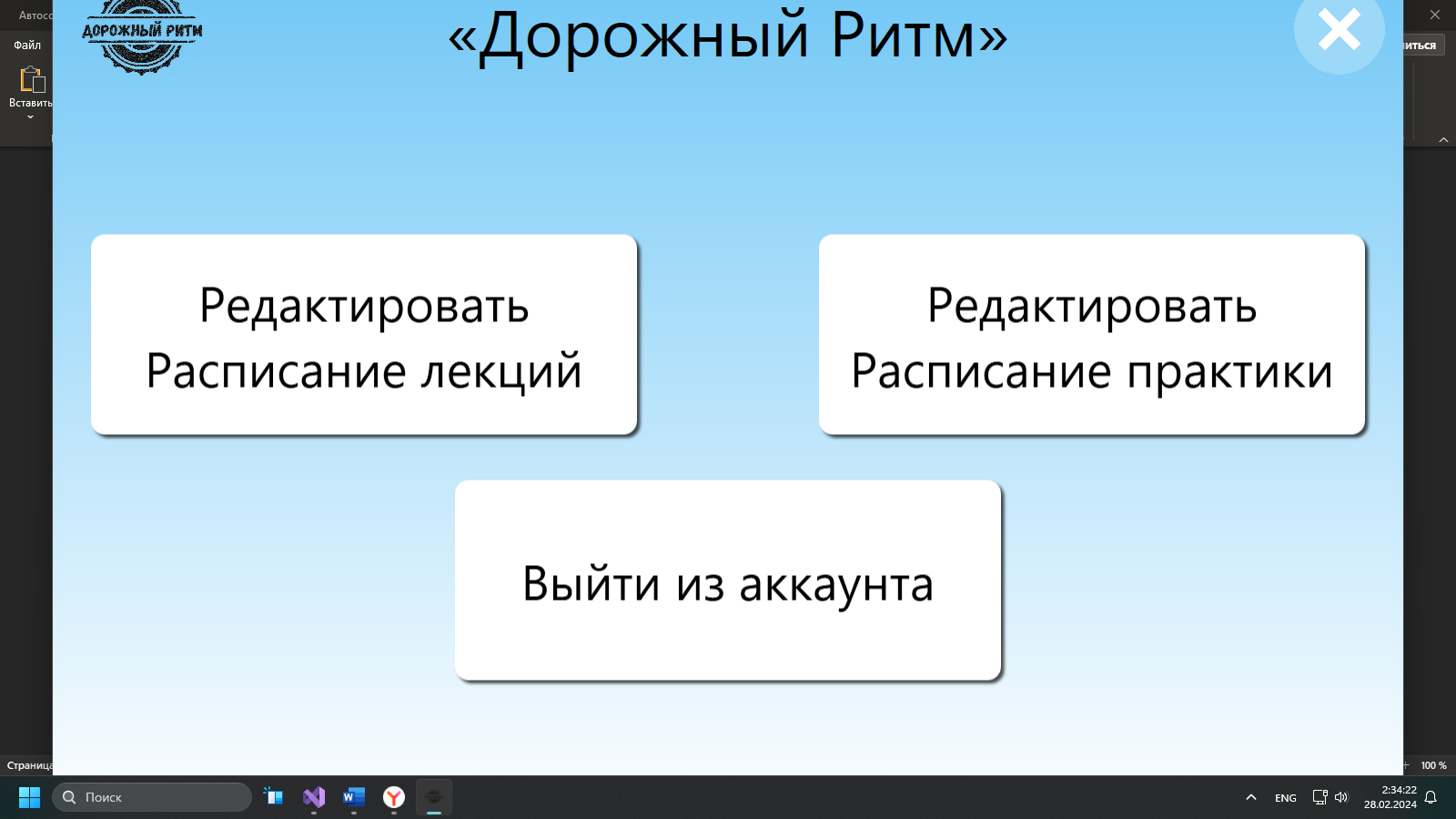


Рисунок 2.12. – «Окно Администратора»

Как сказано выше, у администратора есть функционал редактирования. При нажатии на кнопку «Редактировать расписание лекций» система выдает ему таблицу из базы данных, где администратор и меняет расписания,

что показано на рисунке 2.14.

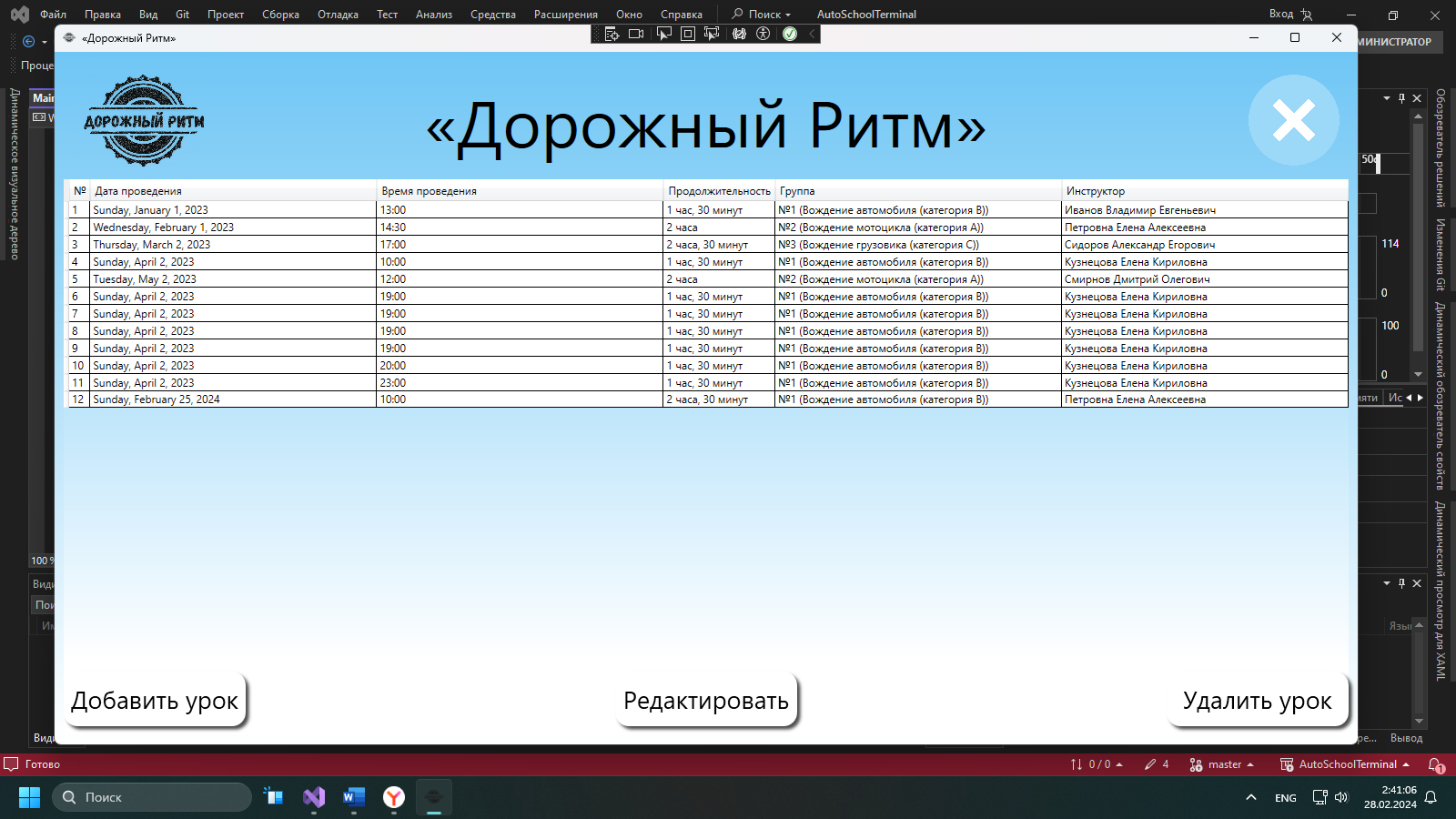


Рисунок 2.14. – «Окно редактирования администратора»

Так же в систему можно войти под ролью инструктора, что показано на рисунке 2.15.

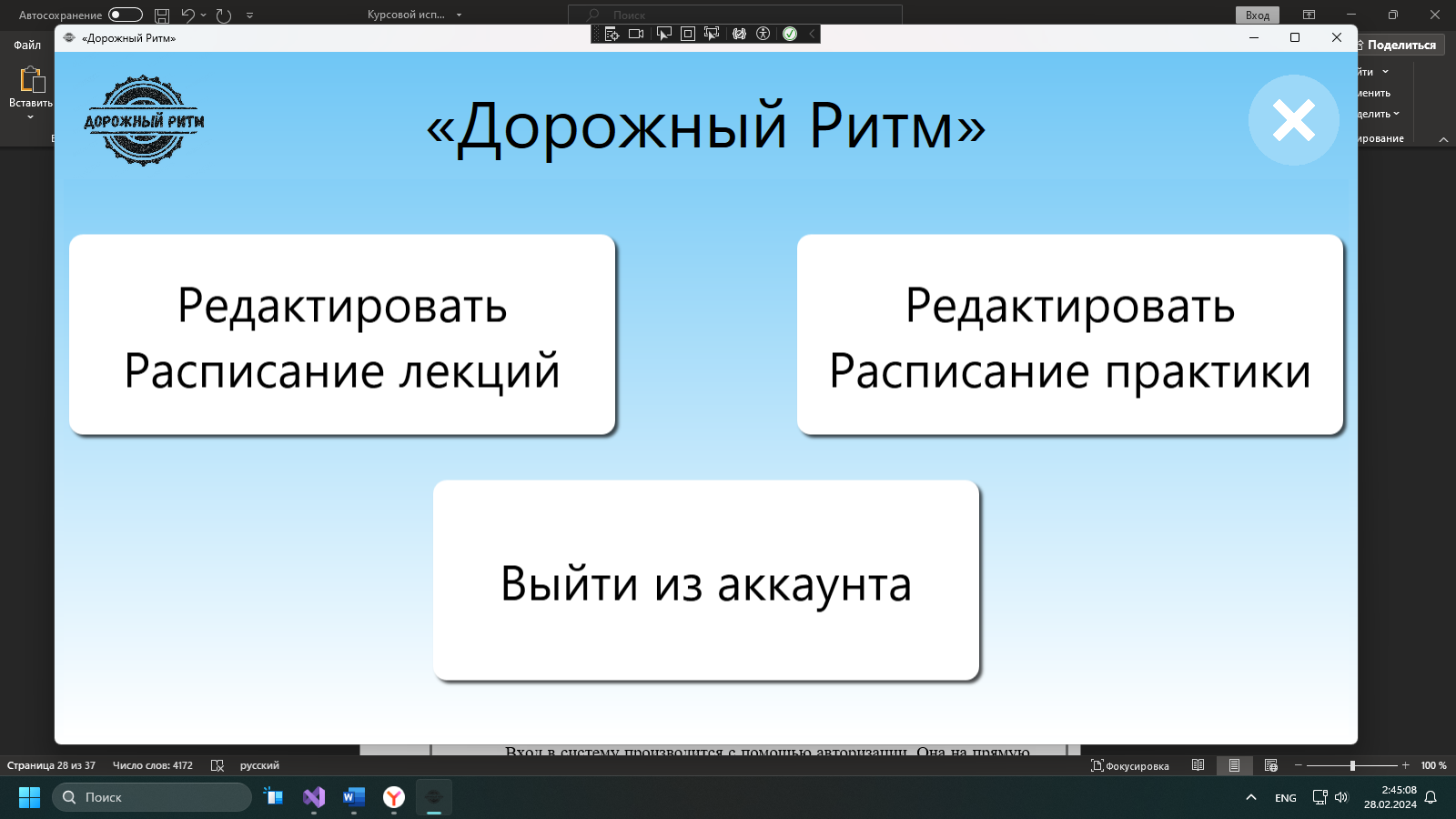


Рисунок 2.15. – «Окно Инструктора»

Инструктор так же как и администратор имеет возможность вносить корректировки в учебный процесс.

Личный кабинет курсантов на данный момент недоступен, но при вводе пароля система выдает уведомление,

показанное на рисунке 2.16.

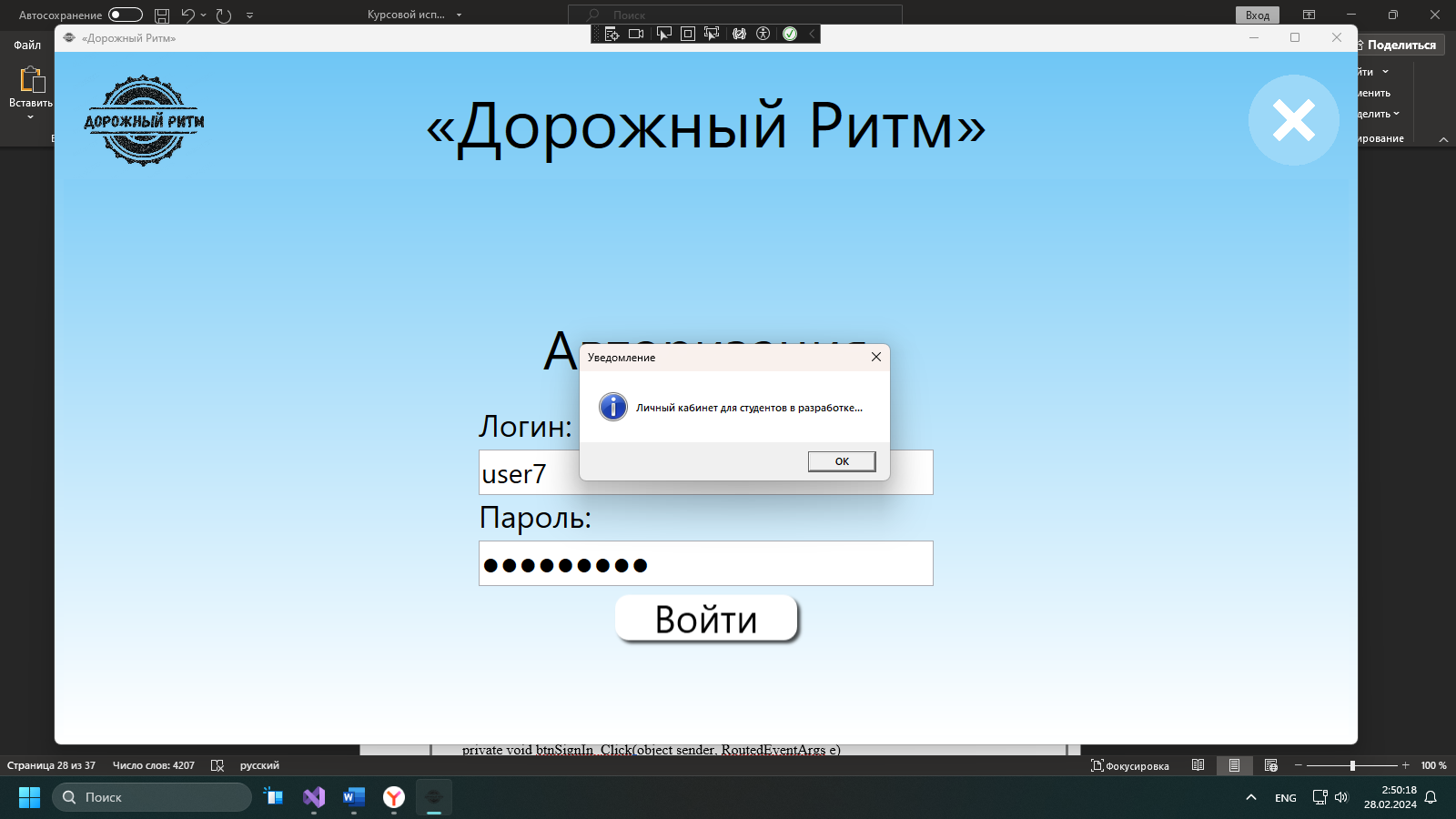


Рисунок 2.16. – «Уведомление при входе курсантом»

Система также выдает уведомление при входе под любой ролью, что показано на рисунках 2.17 и 2.18.

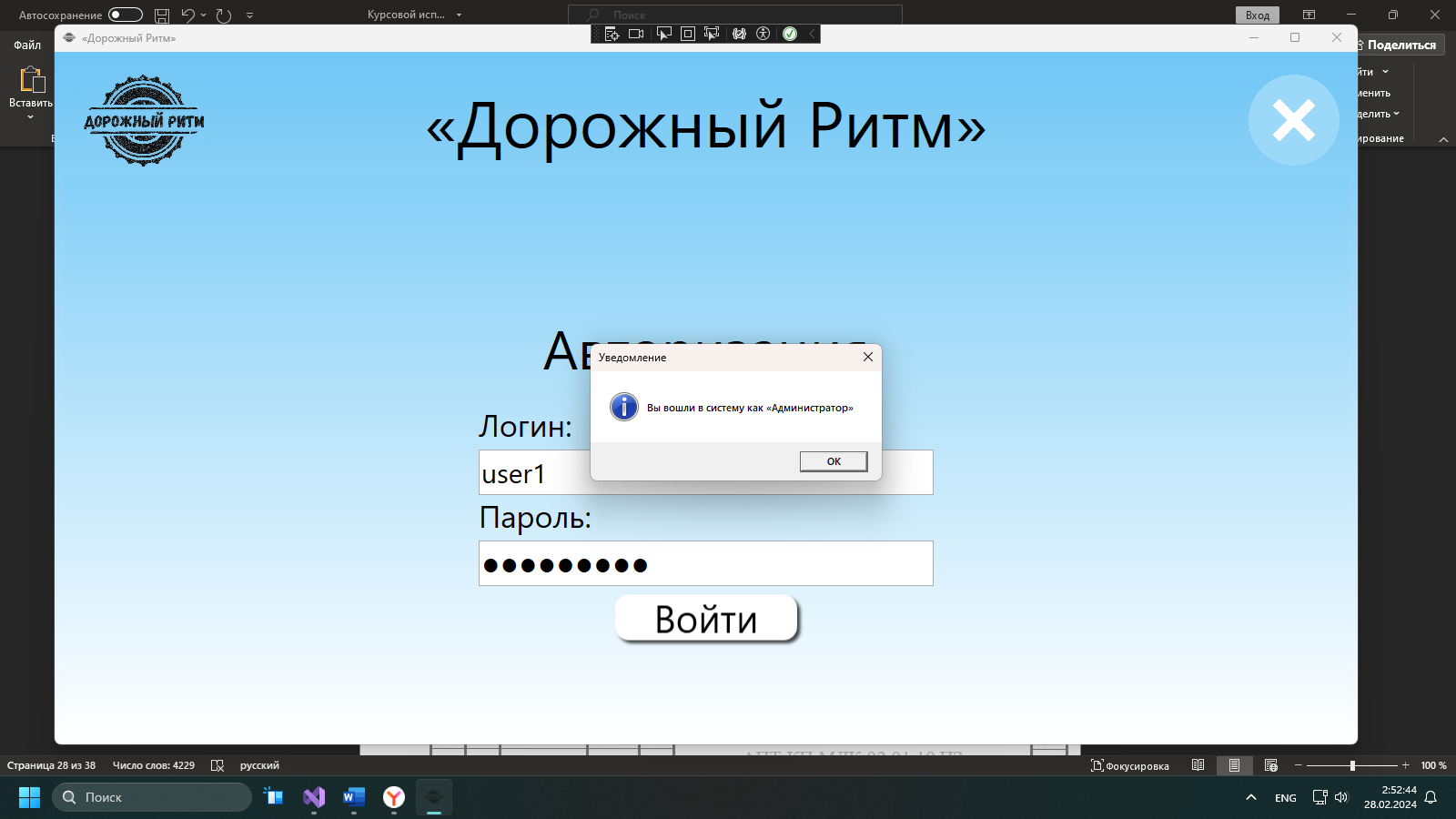


Рисунок 2.17 – «Уведомление при входе Администратором»

Аналогично и для Инструктора

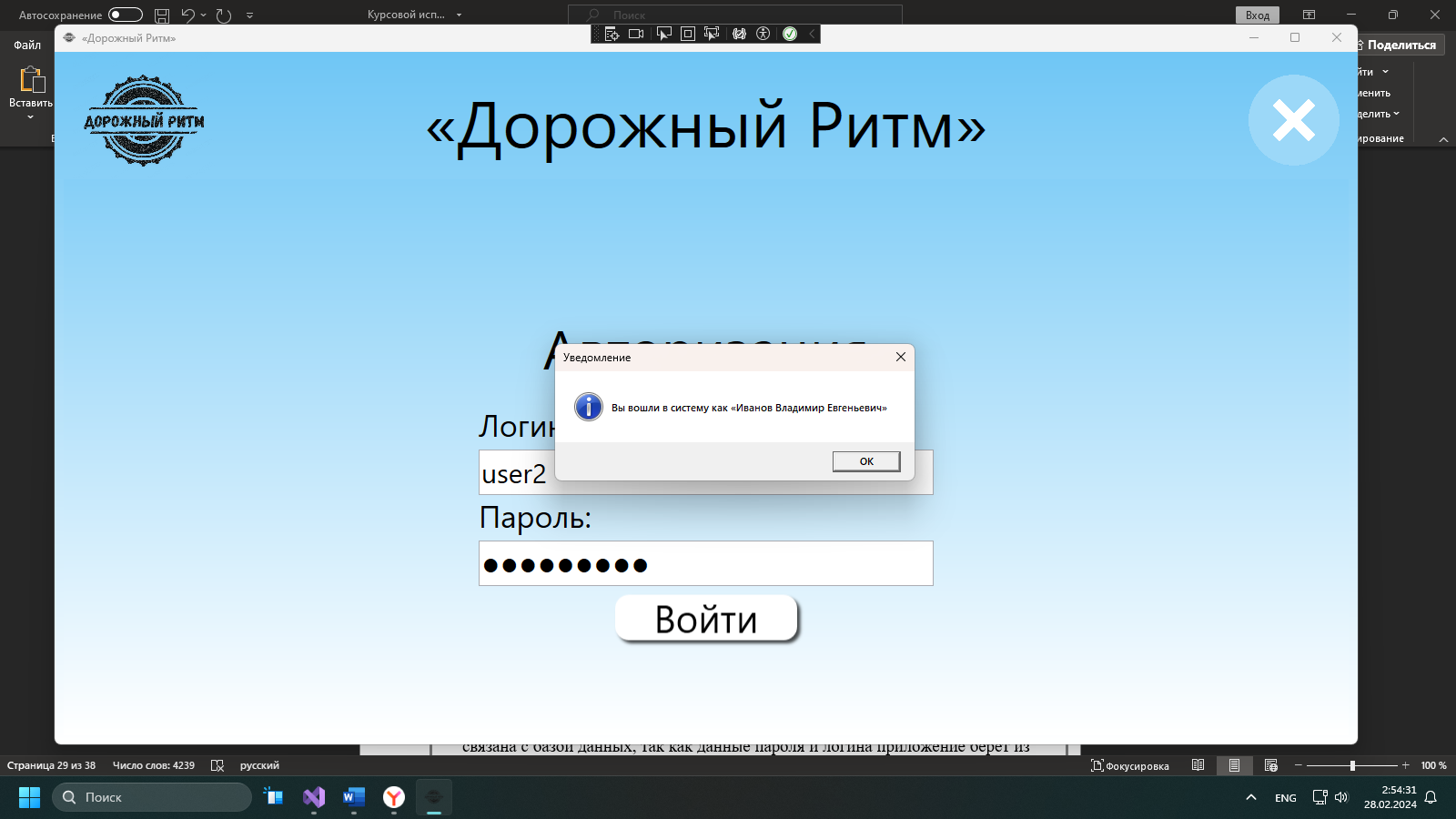


Рисунок 2.18. – «Уведомление при входе Инструктором»

На главной странице мы можем посмотреть подробную информацию об автошколе, нажав на вопросительный знак в правом верхнем углу. Она информирует новых учеников, что показано на рисунке 2.19.

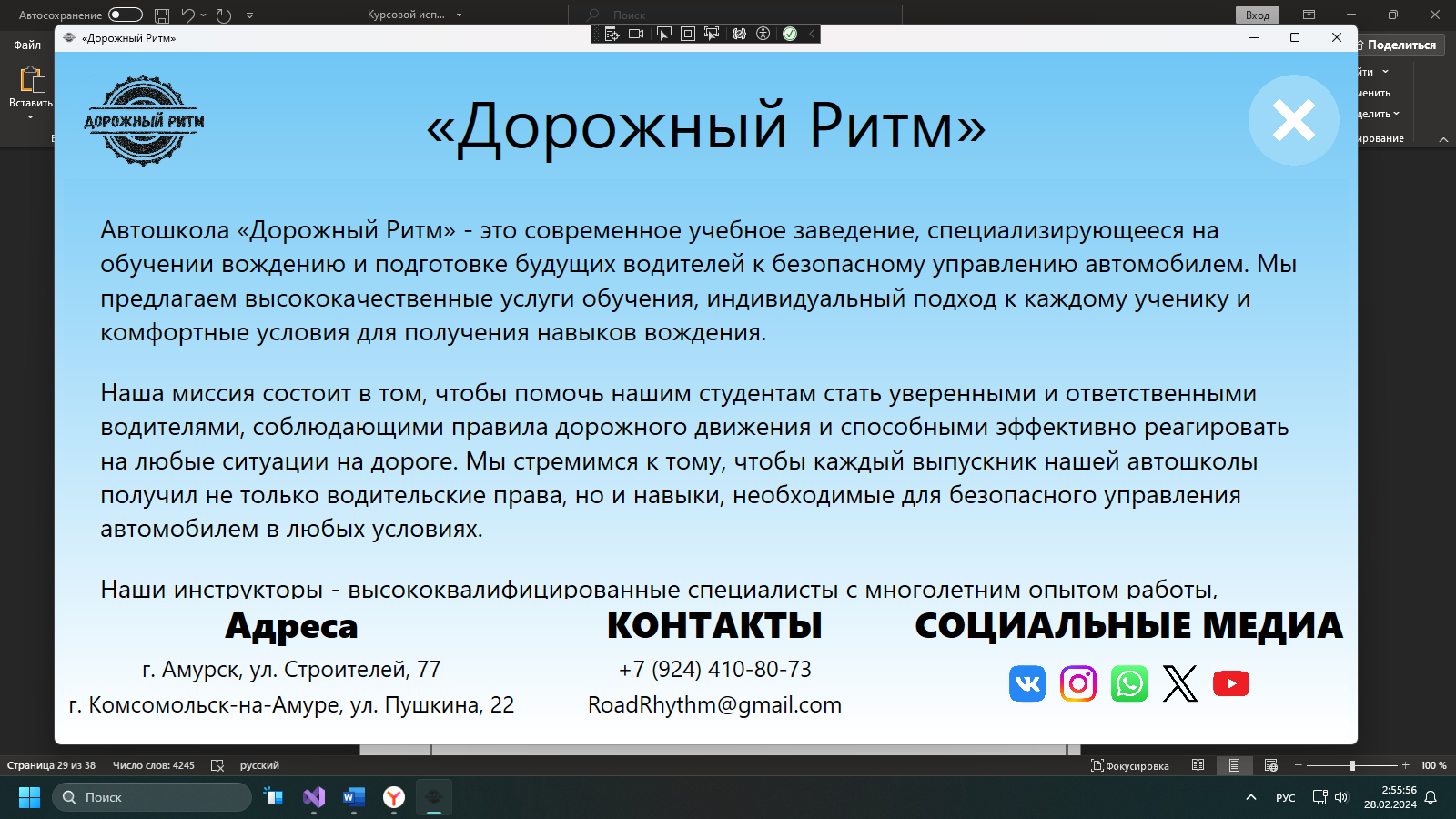


Рисунок 2.19. – «Окно информации об автошколе»

У учеников есть возможность посмотреть расписание, не выжидая часами пока позвонит или напишет инструктор. Ему просто необходимо нажать на кнопку «Расписание», что показано на рисунке 2.20.

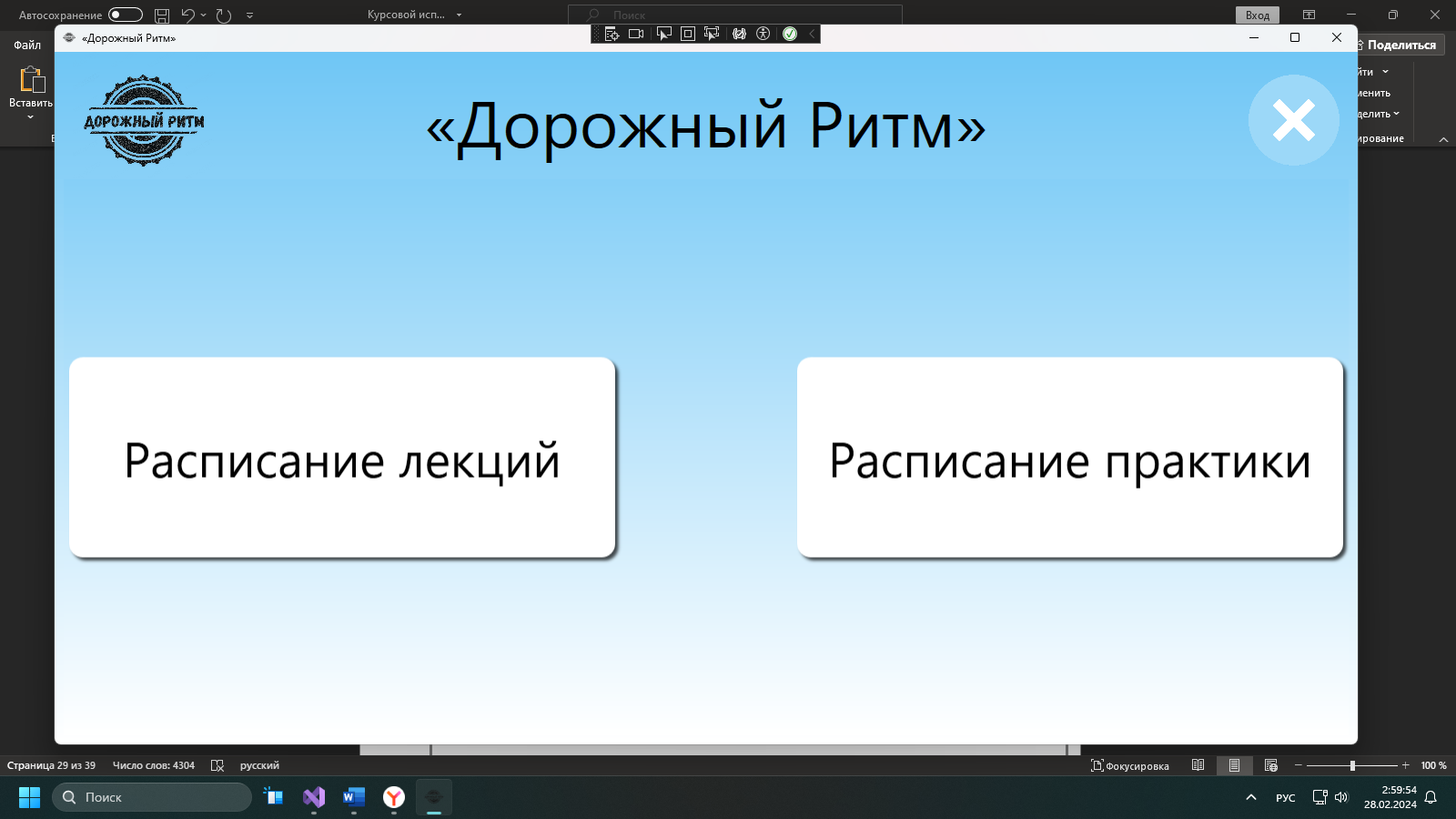


Рисунок 2.20. – «Окно для выбора нужного расписания»

Нажав на кнопку Расписание лекций – курсанту выпадает таблица из базы данных с временем и группой, у которой будет Обучение, что изображено на рисунке 2.21.

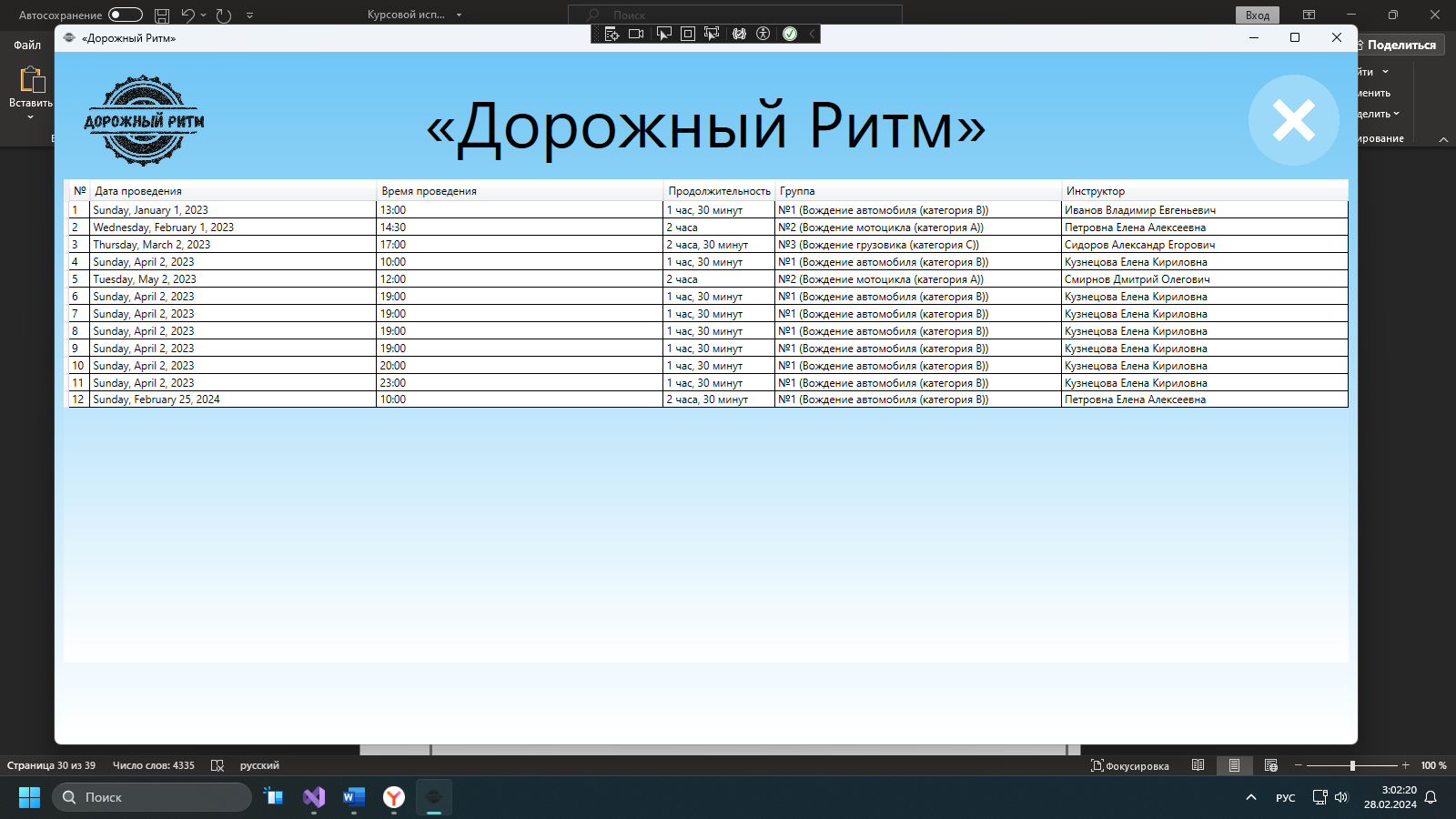


Рисунок 2.21. – «Окно Раписание лекций»

Аналогичная ситуация и при выборе курсантом кнопки с практическим занятием, что изабражено на рисунке 2.22.

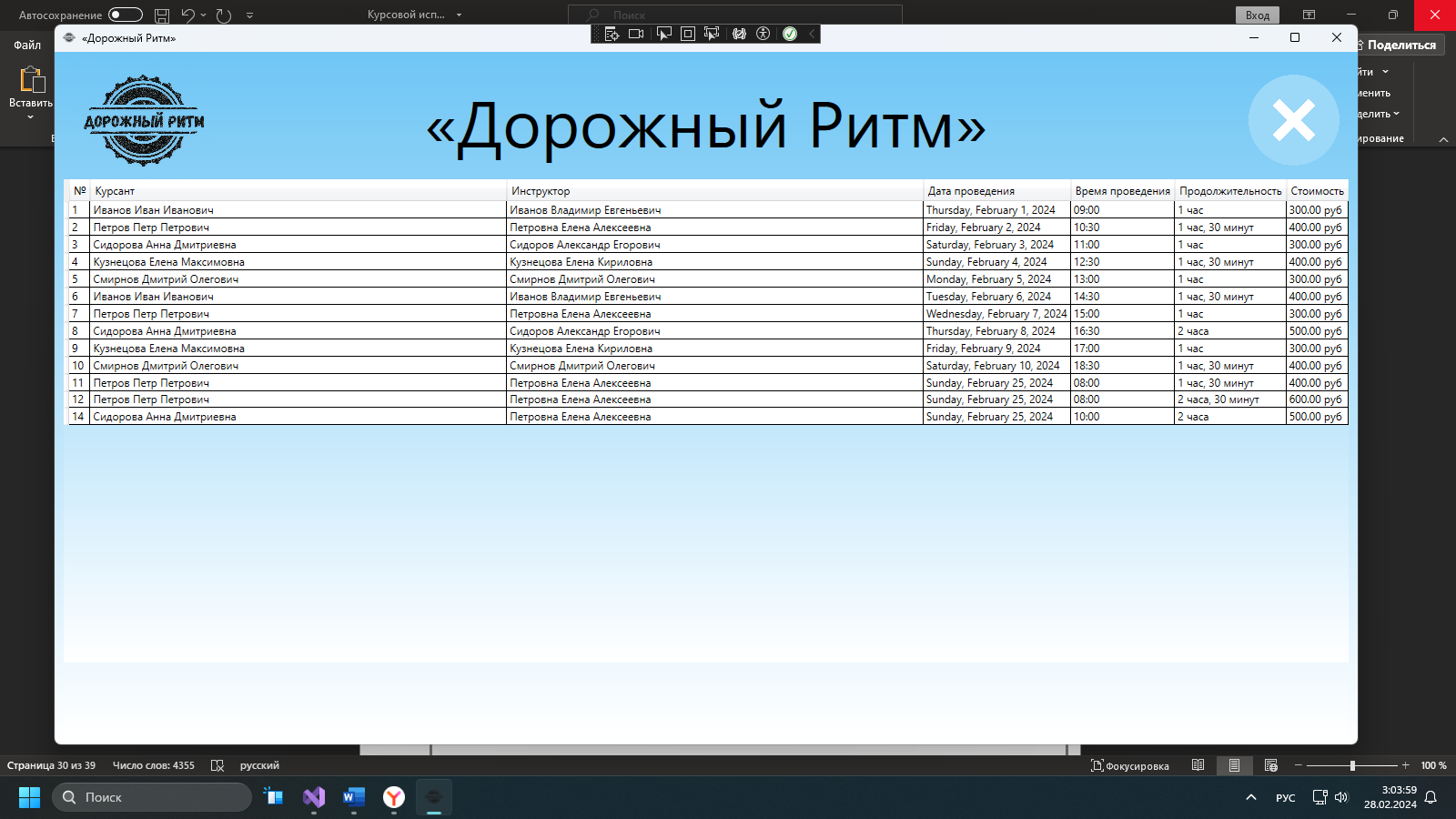


Рисунок 2.22. – «Окно Расписание практики»

Нажав на кнопку Инструкторы, курсант может просмотреть информацию в виде приятных глазу карточек об всех инструкторах автошколы и прочитать контакты. Имеется электронная почта и продолжительность работы в автошколе, что показано на рисунке 2.23.

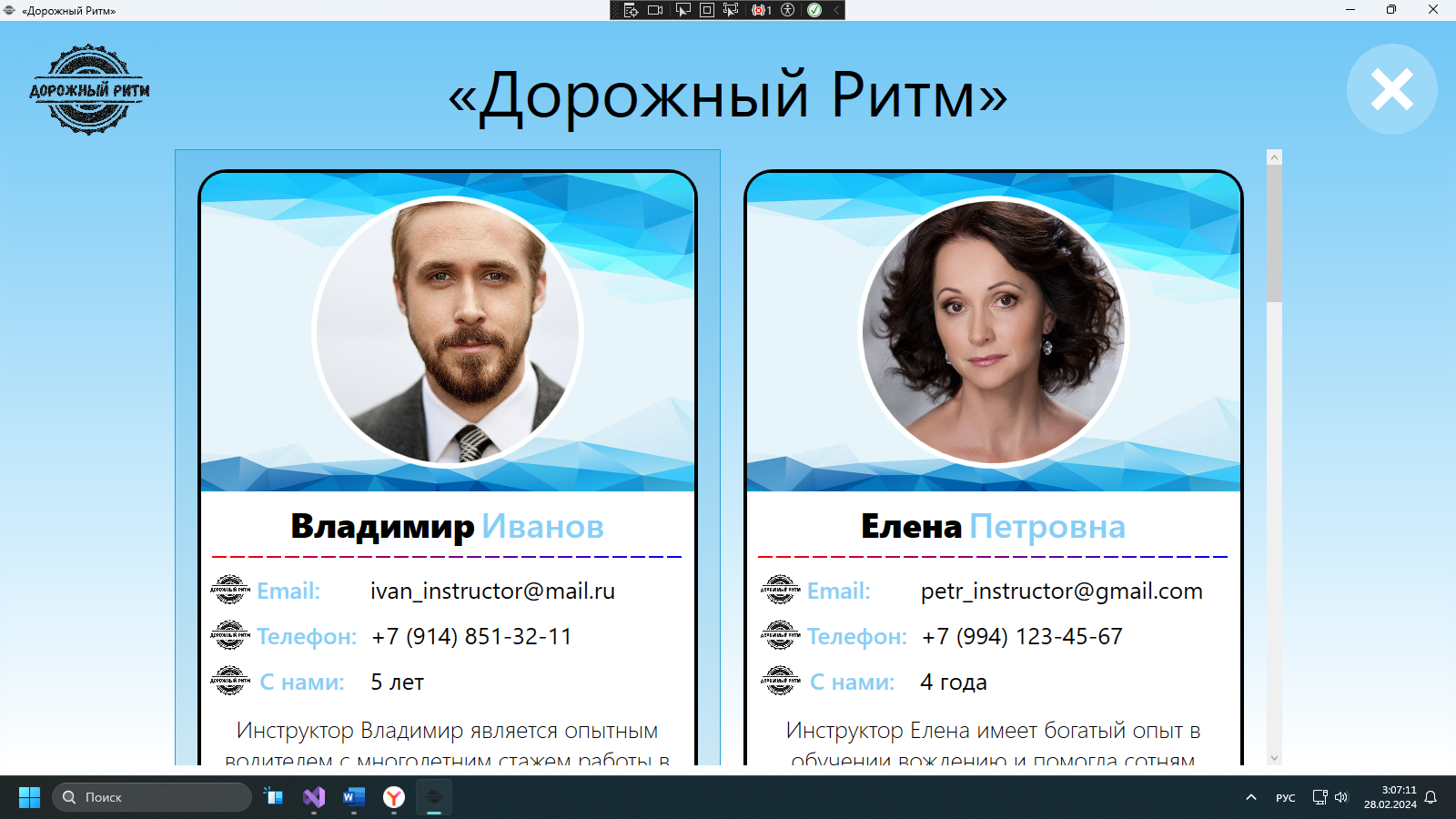


Рисунок 2.23. – «Окно Инструктора»

Так как моё приложение создано только для установки на терминал, функция оплаты за услуги автошколы пока недоступна, что показано на рисунке 2.24.

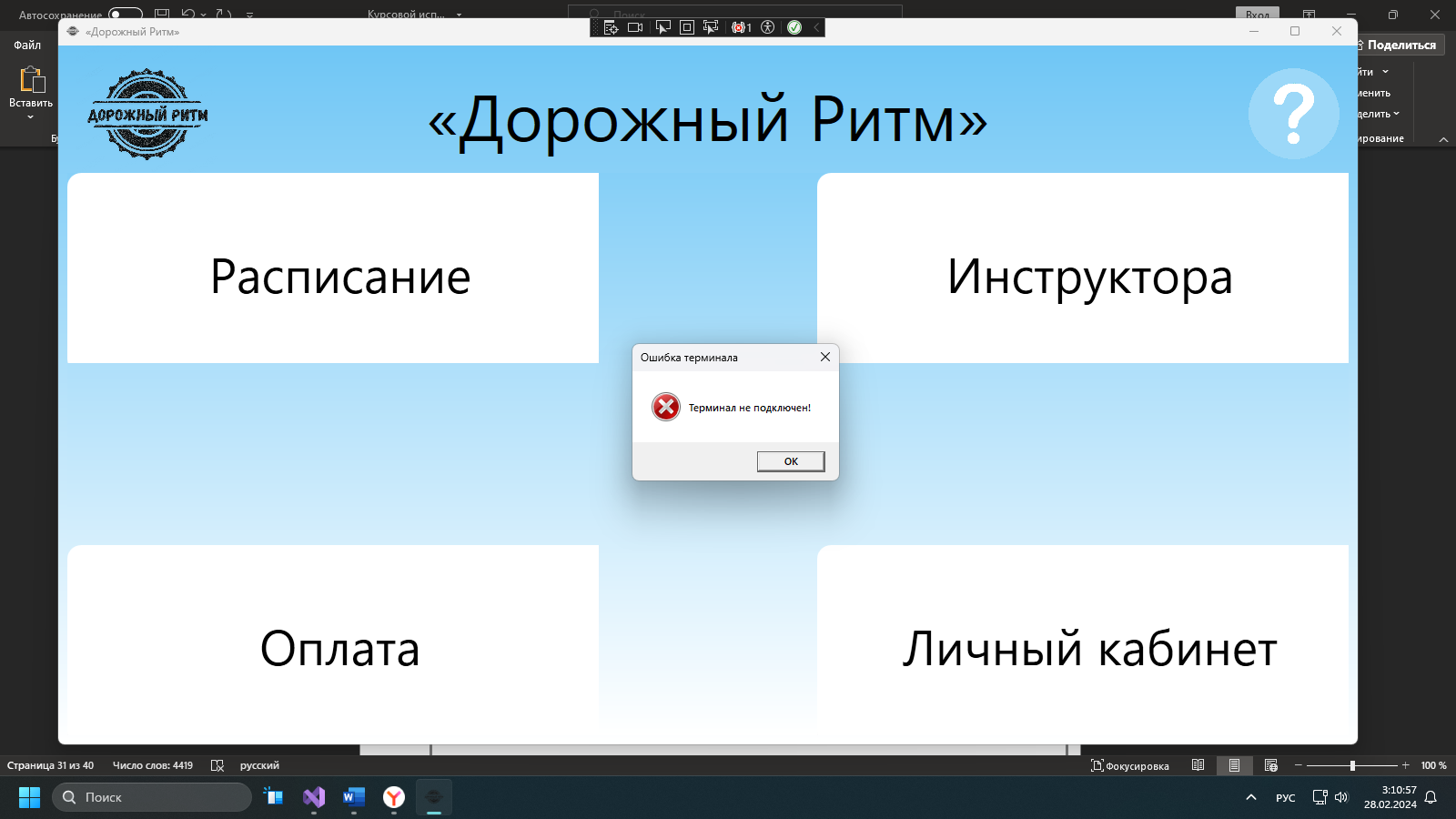


Рисунок 2.24. – «Уведомление раздела об оплате»

# 2.7. Написание программного кода

Вход в систему производится с помощью авторизации. Она на прямую связана с базой данных, так как данные пароля и логина приложение берет из неё, что представлено в листинге 2.1.

Листинг 2.1. «GetContext»

private void btnSignIn\_Click(object sender, RoutedEventArgs e) {StringBuilder errors = new StringBuilder(); if (string.IsNullOrEmpty(txtLogin.Text)) errors.AppendLine("Поле \"Логин\" должно быть заполнено!"); if (string.IsNullOrEmpty(pbPassword.Password)) errors.AppendLine("Поле \"Пароль\" должно быть заполнено!"); if(errors.Length > 0) { MessageBox.Show(errors.ToString(), "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error); pbPassword.Password = string.Empty; } else { Users \_currentUser = DataBaseEntities.GetContext().Users.FirstOrDefault(x => x.Username == txtLogin.Text || x.Password == pbPassword.Password); if (\_currentUser != null) { Manager.AuthUser = \_currentUser; switch (\_currentUser.RoleID) { case 1: // Admin MessageBox.Show($"Вы вошли в систему как «Администратор»", "Уведомление", MessageBoxButton.OK, MessageBoxImage.Information); Manager.MainFrame.Navigate(new InstructorPage()); break; case 2: // Insructor MessageBox.Show($"Вы вошли в систему как «{DataBaseEntities.GetContext().Instructors.FirstOrDefault(x => x.UserID == \_currentUser.UserID).FullName}»", "Уведомление", MessageBoxButton.OK, MessageBoxImage.Information); Manager.MainFrame.Navigate(new InstructorPage()); break; case 3: // Student MessageBox.Show($"Личный кабинет для студентов в разработке...", "Уведомление", MessageBoxButton.OK, MessageBoxImage.Information); Manager.MainFrame.GoBack(); Manager.AuthUser = null; break; default: MessageBox.Show($"Вы вошли в систему под неизвестной ролью!", "Уведомление", MessageBoxButton.OK, MessageBoxImage.Information); break; } } else { MessageBox.Show("Пользователь с такими данными не найден!", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error); } } }

При заполнении расписаний занятий добавление данных в приложении происходит через кнопку сохранить, что показано в листинге 2.2.

Листинг 2.2. «SaveChanges()»

private void btnSave\_Click(object sender, RoutedEventArgs e) { StringBuilder errors = new StringBuilder(); if (cmbStudents.SelectedItem == null) errors.AppendLine("Выберите курсанта!"); if (cmbInsructors.SelectedItem == null) errors.AppendLine("Выберите инструктора!"); if (dpDate.SelectedDate == null) errors.AppendLine("Укажите дату!"); if (string.IsNullOrWhiteSpace(txtBTime.Text)) { errors.AppendLine("Укажите время!"); } else { string timeInput = txtBTime.Text.Trim(); if (!DateTime.TryParseExact(timeInput, "HH:mm", CultureInfo.InvariantCulture, DateTimeStyles.None, out \_)) { errors.AppendLine("Неверный формат времени. Введите время в формате 'чч:мм'."); } else { if (DateTime.ParseExact(timeInput, "HH:mm", CultureInfo.InvariantCulture).TimeOfDay < TimeSpan.FromHours(8) || DateTime.ParseExact(timeInput, "HH:mm", CultureInfo.InvariantCulture).TimeOfDay >= TimeSpan.FromHours(18)) { errors.AppendLine("Время должно быть в диапазоне от 8:00 до 18:00."); } } } if (cmbDuration.SelectedItem == null) errors.AppendLine("Укажите продолжительность!"); if (errors.Length > 0) { MessageBox.Show(errors.ToString(), "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error); return; } if (\_currentDrivingLesson.DrivingLessonID == 0) { try { DataBaseEntities.GetContext().DrivingLessons.Add(\_currentDrivingLesson); } catch (Exception ex) { MessageBox.Show(ex.Message, "Ошибка", MessageBoxButton.OK, MessageBoxImage.Warning); } } try { DataBaseEntities.GetContext().SaveChanges(); MessageBox.Show("Информация сохранена!", "Информация", MessageBoxButton.OK, MessageBoxImage.Information); Manager.MainFrame.GoBack(); } catch (Exception ex) { MessageBox.Show(ex.Message, "Ошибка", MessageBoxButton.OK, MessageBoxImage.Warning); } }

Перед удалением данных подсчитывается кол-во удаляемых строк, данных, а также если они задействованы с другими таблицы с помощь связей об сообщается, что показано в листинге 2.3.

Листинг 2.3. «RemoveRange»

private void btnDel\_Click(object sender, RoutedEventArgs e) { var rowForRemoving = dgLessons.SelectedItems.Cast<Lessons>().ToList(); if(rowForRemoving.Count == 0 ) { MessageBox.Show("Выберите уроки для удаления!", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error); } else { if (MessageBox.Show($"Вы точно хотите удалить следующие {rowForRemoving.Count()} элементов?", "Информация", MessageBoxButton.YesNo, MessageBoxImage.Information) == MessageBoxResult.Yes) { try { DataBaseEntities.GetContext().Lessons.RemoveRange(rowForRemoving); DataBaseEntities.GetContext().SaveChanges(); MessageBox.Show("Данные удалены!", "Информация", MessageBoxButton.OK, MessageBoxImage.Information); DataBaseEntities.GetContext().ChangeTracker.Entries().ToList().ForEach(x => x.Reload()); dgLessons.ItemsSource = DataBaseEntities.GetContext().Lessons.ToList(); } catch (Exception ex) { MessageBox.Show(ex.Message, "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error); } } } }

# 2.8. Защита информации

Защита информации – важный и обязательный пункт при создании программного обеспечения. С этой задачей рано или поздно придется столкнуться каждому разработчику приложений. Каждая составляющая должна быть надежно защищена.

Проверка целостности данных. Все запросы взаимодействия приложения с базой данных проходят через обработчиков, которые осуществляют проверку целостности и корректности данных во избежания каких-либо ошибок.

Безопасность Windows.

Ядро Windows предоставляет разные функции безопасности, которые формируют основу безопасности для всех приложений Windows, включая приложения, созданные с помощью WPF. Windows обеспечивает защиту путем перекомпиляции многих системных библиотек ядра, включая все зависимости WPF, такие как CLR, чтобы снизить вероятность переполнения буфера. В общих чертах, флаг компилятора -GS защищает от некоторых потенциальных переполнений буфера путем добавления специального файла cookie безопасности для защиты адреса функции, которая содержит буферы локальных строк. После возврата функции cookie безопасности сравнивается со своим предыдущим значением. Если значение изменилось, то возможно, произошло переполнение буфера, и процесс будет остановлен с состоянием ошибки. Остановка процесса предотвращает выполнение потенциально вредоносного кода.

Развертывание ClickOnceClickOnce — это комплексная технология развертывания, которая входит в состав .NET Framework и интегрируется с Visual Studio. Приложения, развернутые с помощью ClickOnce, приобретают дополнительный уровень безопасности посредством управления доступом для кода. Приложения, развернутые с помощью ClickOnce, запрашивают разрешения, которые им необходимы. Им предоставляются только разрешения, не расширяющие набор разрешений для зоны, в которой развертывается приложения. Таким образом, уменьшается вероятность повреждений на клиентском компьютере в случае перехвата приложения.

Резервное копирование базы данных (BackUp). Регулярно создаются резервные копии всех данных сайта, включая базы данных, файлы конфигурации и контент. Это обеспечивает возможность восстановления информации в случае ее случайного удаления, атаки злоумышленников или других инцидентов, которые могут привести к потере данных. Резервные копии хранятся на отдельных защищенных серверах или в облачном хранилище с ограниченным доступом только для авторизованных сотрудников. Проводится регулярная проверка работоспособности процесса создания резервных копий и их целостности, чтобы гарантировать их доступность в случае необходимости восстановления данных.

# Заключение

В ходе выполнения курсового проекта был спроектирован модуль информационной системы автошколы для терминала. Также была спроектирована схема и база данных.

В ходе курсового проекта был реализован модуль информационной

системы, который позволит повысить эффективность выполнения работ и оказанию услуг в организации учебного процесса, улучшить работу с клиентами, повысить качество работы.

Также закреплены теоретические знания и практические навыки работы с SQL Server, C# и XAML, умение работать с литературой, анализировать источники, делать обоснованные выводы. Полученные теоретические и практические навыки будут использованы в дальнейшем обучении и, возможно, написании дипломной работы в будущем.

В рамках курсового проекта были выполнены следующие цели:

• автоматизирован процесс представления информации;

• автоматизирован процесс составлении расписания теоретических и практические занятий;

• повышен уровень доступности к предоставленной информации;

• упрощен процесс получения информации;

И задачи:

•Определены требования к программному продукту.

•Разработан прототип приложения.

•Реализован приложение в выбранной среде IDE.

•Создана сборка для установки.

•Проект размещен на веб-сервисе GitHub или аналогичном – https://github.com/MariaAPT2/AutoSchoolTerminal1

# Список использованных источников

1.Тузовский, А. Ф. Проектирование и разработка web-приложений: учебное пособие для вузов / А. Ф. Тузовский. — Москва: Издательство Юрайт, 2023. — 218 с. — (Высшее образование). — ISBN 978-5-534-00515-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/512113> (дата обращения: 10.02.2024).

2.Сысолетин, Е. Г. Разработка интернет-приложений: учебное пособие для вузов / Е. Г. Сысолетин, С. Д. Ростунцев ; под научной редакцией Л. Г. Доросинского. — Москва: Издательство Юрайт, 2022. — 90 с. — (Высшее образование). — ISBN 978-5-9916-9975-4. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/492224> (дата обращения: 05.02.2024).

3.Сысолетин, Е. Г. Разработка интернет-приложений: учебное пособие для среднего профессионального образования / Е. Г. Сысолетин, С. Д. Ростунцев. — Москва: Издательство Юрайт, 2023. — 90 с. — (Профессиональное образование). — ISBN 978-5-534-10015-0. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/517538> (дата обращения: 05.02.2024).

4.Полуэктова, Н. Р. Разработка веб-приложений: учебное пособие для среднего профессионального образования / Н. Р. Полуэктова. — Москва: Издательство Юрайт, 2023. — 204 с. — (Профессиональное образование). — ISBN 978-5-534-14744-5. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/519727> (дата обращения: 05.02.2024).

5.Полуэктова, Н. Р. Разработка веб-приложений: учебное пособие для вузов / Н. Р. Полуэктова. — Москва: Издательство Юрайт, 2023. — 204 с. — (Высшее образование). — ISBN 978-5-534-13715-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/519714> (дата обращения: 05.02.2024).

# Приложения

**Приложение А**

**Листинги программного кода**

**App.xaml**

<Application.Resources> <Style TargetType="TextBlock"> <Setter Property="FontFamily" Value="Segoe UI"/> <Setter Property="HorizontalAlignment" Value="Center"/> <Setter Property="VerticalAlignment" Value="Center"/> <Setter Property="TextAlignment" Value="Center"/> </Style> <Style x:Key="ImageInformationButtonStyle" TargetType="Image"> <!--Animation Click...--> </Style> <Style x:Key="HeaderFooterText" TargetType="TextBlock"> <Setter Property="FontFamily" Value="Segoe UI"/> <Setter Property="HorizontalAlignment" Value="Center"/> <Setter Property="VerticalAlignment" Value="Center"/> <Setter Property="FontSize" Value="40"/> <Setter Property="FontWeight" Value="ExtraBold"/> </Style> <Style x:Key="ContactsText" TargetType="TextBlock"> <Setter Property="FontFamily" Value="Segoe UI"/> <Setter Property="HorizontalAlignment" Value="Center"/> <Setter Property="VerticalAlignment" Value="Center"/> <Setter Property="FontSize" Value="25"/> <Setter Property="Margin" Value="0 6 0 0"/> </Style> <Style x:Key="ImageLinksStyles" TargetType="Image"> <Setter Property="HorizontalAlignment" Value="Center"/> <Setter Property="Width" Value="40"/> <Setter Property="Height" Value="40"/> <Setter Property="Cursor" Value="Hand"/> <Setter Property="Margin" Value="8 20 8 0"/> <Setter Property="VerticalAlignment" Value="Center"/> </Style> <Style x:Key="DescriptionOfAutoScholText" TargetType="TextBlock"> <Setter Property="HorizontalAlignment" Value="Left"/> <Setter Property="VerticalAlignment" Value="Center"/> <Setter Property="TextAlignment" Value="Left"/> <Setter Property="Width" Value="auto"/> <Setter Property="Height" Value="auto"/> <Setter Property="Margin" Value="40 15 40 15"/> <Setter Property="TextWrapping" Value="Wrap"/> <Setter Property="FontSize" Value="28"/> <Setter Property="VerticalAlignment" Value="Center"/> </Style> <Style x:Key="MainButtonStyle" TargetType="Button"> <Setter Property="Width" Value="600"/> <Setter Property="Height" Value="220"/> <Setter Property="FontSize" Value="54"/> <Setter Property="Background" Value="White"/> <Setter Property="Foreground" Value="Black"/> <Setter Property="BorderThickness" Value="0"/> <Setter Property="Padding" Value="10"/> <Setter Property="Cursor" Value="Hand"/> <Setter Property="HorizontalContentAlignment" Value="Center"/> <Setter Property="VerticalContentAlignment" Value="Center"/> <Setter Property="Template"> <Setter.Value> <ControlTemplate TargetType="Button"> <Border Name="MainBorder" Background="{TemplateBinding Background}" CornerRadius="15" BorderThickness="{TemplateBinding BorderThickness}" BorderBrush="{TemplateBinding BorderBrush}"> <Border.Effect> <DropShadowEffect ShadowDepth="5" Direction="315" Color="Black" Opacity="0.7"/> </Border.Effect> <ContentPresenter HorizontalAlignment="Center" VerticalAlignment="Center"/> </Border> <ControlTemplate.Triggers> <Trigger Property="IsPressed" Value="True"> <Setter TargetName="MainBorder" Property="Margin" Value="5,5,0,0"/> </Trigger> </ControlTemplate.Triggers> </ControlTemplate> </Setter.Value> </Setter> </Style> <Style x:Key="InputAuthDataTextBox" TargetType="TextBox"> <Setter Property="FontFamily" Value="Segoe UI"/> <Setter Property="FontSize" Value="30"/> <Setter Property="Width" Value="500"/> <Setter Property="Height" Value="50"/> <Setter Property="VerticalContentAlignment" Value="Center"/> <Setter Property="Foreground" Value="Black"/> </Style> <Style x:Key="InputAuthDataPasswordBox" TargetType="PasswordBox"> <Setter Property="FontFamily" Value="Segoe UI"/> <Setter Property="FontSize" Value="34"/> <Setter Property="Width" Value="500"/> <Setter Property="Height" Value="50"/> <Setter Property="VerticalContentAlignment" Value="Top"/> <Setter Property="Foreground" Value="Black"/> </Style> <Style x:Key="TextBlockAuthData" TargetType="TextBlock"> <Setter Property="FontFamily" Value="Segoe UI"/> <Setter Property="FontSize" Value="34"/> <Setter Property="Width" Value="500"/> <Setter Property="Height" Value="50"/> <Setter Property="Margin" Value="0"/> <Setter Property="Foreground" Value="Black"/> </Style> <Style x:Key="DataStyle" TargetType="DataGrid"> <Setter Property="AutoGenerateColumns" Value="False"/> <Setter Property="CanUserAddRows" Value="False"/> <Setter Property="CanUserDeleteRows" Value="False"/> <Setter Property="IsReadOnly" Value="True"/> <Setter Property="Margin" Value="0"/> <Setter Property="BorderThickness" Value="0"/> </Style> <Style x:Key="TextInputColumns" TargetType="TextBlock"> <Setter Property="FontFamily" Value="Segoe UI"/> <Setter Property="FontSize" Value="32"/> <Setter Property="Height" Value="50"/> <Setter Property="HorizontalAlignment" Value="Center"/> <Setter Property="VerticalAlignment" Value="Bottom"/> <Setter Property="Margin" Value="0 15 0 0"/> <Setter Property="Foreground" Value="Black"/> </Style> <Style x:Key="SelectInputColumns" TargetType="ComboBox"> <Setter Property="FontFamily" Value="Segoe UI"/> <Setter Property="FontSize" Value="24"/> <Setter Property="Height" Value="50"/> <Setter Property="Width" Value="400"/> <Setter Property="HorizontalAlignment" Value="Center"/> <Setter Property="VerticalContentAlignment" Value="Center"/> <Setter Property="Foreground" Value="Black"/> </Style> <Style TargetType="DatePicker"> <Setter Property="FontFamily" Value="Segoe UI"/> <Setter Property="FontSize" Value="24"/> <Setter Property="Height" Value="50"/> <Setter Property="Width" Value="400"/> <Setter Property="HorizontalAlignment" Value="Center"/> <Setter Property="VerticalContentAlignment" Value="Top"/> <Setter Property="Foreground" Value="Black"/> <Setter Property="IsTodayHighlighted" Value="True"/> <Setter Property="SelectedDateFormat" Value="Short"/> <Setter Property="Background" Value="Transparent"/> <Setter Property="Padding" Value="2"/> <Setter Property="BorderBrush"> <Setter.Value> <LinearGradientBrush EndPoint=".5,0" StartPoint=".5,1"> <GradientStop Color="#FFA3AEB9" Offset="0"/> <GradientStop Color="#FF8399A9" Offset="0.375"/> <GradientStop Color="#FF718597" Offset="0.375"/> <GradientStop Color="#FF617584" Offset="1"/> </LinearGradientBrush> </Setter.Value> </Setter> <Setter Property="BorderThickness" Value="1"/> <Setter Property="HorizontalContentAlignment" Value="Stretch"/> <Setter Property="Template"> <Setter.Value> <ControlTemplate TargetType="{x:Type DatePicker}"> <Border Background="{TemplateBinding Background}" BorderThickness="{TemplateBinding BorderThickness}" BorderBrush="{TemplateBinding BorderBrush}" Padding="{TemplateBinding Padding}"> <VisualStateManager.VisualStateGroups> <VisualStateGroup x:Name="CommonStates"> <VisualState x:Name="Normal"/> <VisualState x:Name="Disabled"> <Storyboard> <DoubleAnimation Duration="0" Storyboard.TargetName="PART\_DisabledVisual" Storyboard.TargetProperty="Opacity" To="1"/> </Storyboard> </VisualState> </VisualStateGroup> </VisualStateManager.VisualStateGroups> <Grid x:Name="PART\_Root" HorizontalAlignment="{TemplateBinding HorizontalContentAlignment}" VerticalAlignment="{TemplateBinding VerticalContentAlignment}"> <Grid.Resources> <SolidColorBrush x:Key="DisabledBrush" Color="#A5FFFFFF"/> <ControlTemplate x:Key="DropDownButtonTemplate" TargetType="{x:Type Button}"> <Grid> <VisualStateManager.VisualStateGroups> <VisualStateGroup x:Name="CommonStates"> <VisualStateGroup.Transitions> <VisualTransition GeneratedDuration="0"/> <VisualTransition GeneratedDuration="0:0:0.1" To="MouseOver"/> <VisualTransition GeneratedDuration="0:0:0.1" To="Pressed"/> </VisualStateGroup.Transitions> <VisualState x:Name="Normal"/> <VisualState x:Name="MouseOver"> <Storyboard> <ColorAnimation Duration="0" Storyboard.TargetName="Background" Storyboard.TargetProperty="(Border.Background).(SolidColorBrush.Color)" To="#FF448DCA"/> <ColorAnimationUsingKeyFrames BeginTime="0" Duration="00:00:00.001" Storyboard.TargetName="BackgroundGradient" Storyboard.TargetProperty="(Border.Background).(GradientBrush.GradientStops)[3].(GradientStop.Color)"> <SplineColorKeyFrame KeyTime="0" Value="#7FFFFFFF"/> </ColorAnimationUsingKeyFrames> <ColorAnimationUsingKeyFrames BeginTime="0" Duration="00:00:00.001" Storyboard.TargetName="BackgroundGradient" Storyboard.TargetProperty="(Border.Background).(GradientBrush.GradientStops)[2].(GradientStop.Color)"> <SplineColorKeyFrame KeyTime="0" Value="#CCFFFFFF"/> </ColorAnimationUsingKeyFrames> <ColorAnimationUsingKeyFrames BeginTime="0" Duration="00:00:00.001" Storyboard.TargetName="BackgroundGradient" Storyboard.TargetProperty="(Border.Background).(GradientBrush.GradientStops)[1].(GradientStop.Color)"> <SplineColorKeyFrame KeyTime="0" Value="#F2FFFFFF"/> </ColorAnimationUsingKeyFrames> </Storyboard> </VisualState> <VisualState x:Name="Pressed"> <Storyboard> <ColorAnimationUsingKeyFrames BeginTime="0" Duration="00:00:00.001" Storyboard.TargetName="Background" Storyboard.TargetProperty="(Border.Background).(SolidColorBrush.Color)"> <SplineColorKeyFrame KeyTime="0" Value="#FF448DCA"/> </ColorAnimationUsingKeyFrames> <DoubleAnimationUsingKeyFrames BeginTime="0" Duration="00:00:00.001" Storyboard.TargetName="Highlight" Storyboard.TargetProperty="(UIElement.Opacity)"> <SplineDoubleKeyFrame KeyTime="0" Value="1"/> </DoubleAnimationUsingKeyFrames> <ColorAnimationUsingKeyFrames BeginTime="0" Duration="00:00:00.001" Storyboard.TargetName="BackgroundGradient" Storyboard.TargetProperty="(Border.Background).(GradientBrush.GradientStops)[1].(GradientStop.Color)"> <SplineColorKeyFrame KeyTime="0" Value="#EAFFFFFF"/> </ColorAnimationUsingKeyFrames> <ColorAnimationUsingKeyFrames BeginTime="0" Duration="00:00:00.001" Storyboard.TargetName="BackgroundGradient" Storyboard.TargetProperty="(Border.Background).(GradientBrush.GradientStops)[2].(GradientStop.Color)"> <SplineColorKeyFrame KeyTime="0" Value="#C6FFFFFF"/> </ColorAnimationUsingKeyFrames> <ColorAnimationUsingKeyFrames BeginTime="0" Duration="00:00:00.001" Storyboard.TargetName="BackgroundGradient" Storyboard.TargetProperty="(Border.Background).(GradientBrush.GradientStops)[3].(GradientStop.Color)"> <SplineColorKeyFrame KeyTime="0" Value="#6BFFFFFF"/> </ColorAnimationUsingKeyFrames> <ColorAnimationUsingKeyFrames BeginTime="0" Duration="00:00:00.001" Storyboard.TargetName="BackgroundGradient" Storyboard.TargetProperty="(Border.Background).(GradientBrush.GradientStops)[0].(GradientStop.Color)"> <SplineColorKeyFrame KeyTime="0" Value="#F4FFFFFF"/> </ColorAnimationUsingKeyFrames> </Storyboard> </VisualState> <VisualState x:Name="Disabled"/> </VisualStateGroup> </VisualStateManager.VisualStateGroups> <Grid Background="#11FFFFFF" FlowDirection="LeftToRight" HorizontalAlignment="Center" Height="18" Margin="0" VerticalAlignment="Center" Width="19"> <Grid.ColumnDefinitions> <ColumnDefinition Width="20\*"/> <ColumnDefinition Width="20\*"/> <ColumnDefinition Width="20\*"/> <ColumnDefinition Width="20\*"/> </Grid.ColumnDefinitions> <Grid.RowDefinitions> <RowDefinition Height="23\*"/> <RowDefinition Height="19\*"/> <RowDefinition Height="19\*"/> <RowDefinition Height="19\*"/> </Grid.RowDefinitions> <Border x:Name="Highlight" BorderThickness="1" BorderBrush="#FF45D6FA" CornerRadius="0,0,1,1" Grid.ColumnSpan="4" Margin="-1" Opacity="0" Grid.RowSpan="4" Grid.Row="0"/> <Border x:Name="Background" Background="#FF1F3B53" BorderThickness="1" BorderBrush="#FFFFFFFF" CornerRadius=".5" Grid.ColumnSpan="4" Margin="0,-1,0,0" Opacity="1" Grid.RowSpan="3" Grid.Row="1"/> <Border x:Name="BackgroundGradient" BorderThickness="1" BorderBrush="#BF000000" CornerRadius=".5" Grid.ColumnSpan="4" Margin="0,-1,0,0" Opacity="1" Grid.RowSpan="3" Grid.Row="1"> <Border.Background> <LinearGradientBrush EndPoint=".7,1" StartPoint=".7,0"> <GradientStop Color="#FFFFFFFF" Offset="0"/> <GradientStop Color="#F9FFFFFF" Offset="0.375"/> <GradientStop Color="#E5FFFFFF" Offset="0.625"/> <GradientStop Color="#C6FFFFFF" Offset="1"/> </LinearGradientBrush> </Border.Background> </Border> <Rectangle Grid.ColumnSpan="4" Grid.RowSpan="1" StrokeThickness="1"> <Rectangle.Fill> <LinearGradientBrush EndPoint="0.3,-1.1" StartPoint="0.46,1.6"> <GradientStop Color="#FF4084BD"/> <GradientStop Color="#FFAFCFEA" Offset="1"/> </LinearGradientBrush> </Rectangle.Fill> <Rectangle.Stroke> <LinearGradientBrush EndPoint="0.48,-1" StartPoint="0.48,1.25"> <GradientStop Color="#FF494949"/> <GradientStop Color="#FF9F9F9F" Offset="1"/> </LinearGradientBrush> </Rectangle.Stroke> </Rectangle> <Path Grid.ColumnSpan="4" Grid.Column="0" Data="M11.426758,8.4305077 L11.749023,8.4305077 L11.749023,16.331387 L10.674805,16.331387 L10.674805,10.299648 L9.0742188,11.298672 L9.0742188,10.294277 C9.4788408,10.090176 9.9094238,9.8090878 10.365967,9.4510155 C10.82251,9.0929432 11.176106,8.7527733 11.426758,8.4305077 z M14.65086,8.4305077 L18.566387,8.4305077 L18.566387,9.3435936 L15.671368,9.3435936 L15.671368,11.255703 C15.936341,11.058764 16.27293,10.960293 16.681133,10.960293 C17.411602,10.960293 17.969301,11.178717 18.354229,11.615566 C18.739157,12.052416 18.931622,12.673672 18.931622,13.479336 C18.931622,15.452317 18.052553,16.438808 16.294415,16.438808 C15.560365,16.438808 14.951641,16.234707 14.468243,15.826504 L14.881817,14.929531 C15.368796,15.326992 15.837872,15.525723 16.289043,15.525723 C17.298809,15.525723 17.803692,14.895514 17.803692,13.635098 C17.803692,12.460618 17.305971,11.873379 16.310528,11.873379 C15.83071,11.873379 15.399232,12.079271 15.016094,12.491055 L14.65086,12.238613 z" Fill="#FF2F2F2F" HorizontalAlignment="Center" Margin="4,3,4,3" RenderTransformOrigin="0.5,0.5" Grid.RowSpan="3" Grid.Row="1" Stretch="Fill" VerticalAlignment="Center"/> <Ellipse Grid.ColumnSpan="4" Fill="#FFFFFFFF" HorizontalAlignment="Center" Height="3" StrokeThickness="0" VerticalAlignment="Center" Width="3"/> <Border x:Name="DisabledVisual" BorderThickness="1" BorderBrush="#B2FFFFFF" CornerRadius="0,0,.5,.5" Grid.ColumnSpan="4" Opacity="0" Grid.RowSpan="4" Grid.Row="0"/> </Grid> </Grid> </ControlTemplate> </Grid.Resources> <Grid.ColumnDefinitions> <ColumnDefinition Width="\*"/> <ColumnDefinition Width="Auto"/> </Grid.ColumnDefinitions> <!-- This ViewBox allows to resize the icon automatically to fill the full height of the DatePicker --> <Viewbox Grid.Column="1" HorizontalAlignment="Left" Margin="3,0,3,0" Grid.Row="0" VerticalAlignment="Top"> <Button x:Name="PART\_Button" Focusable="False" Foreground="{TemplateBinding Foreground}" Template="{StaticResource DropDownButtonTemplate}" /> </Viewbox> <DatePickerTextBox x:Name="PART\_TextBox" Grid.Column="0" Focusable="{TemplateBinding Focusable}" HorizontalContentAlignment="Stretch" Grid.Row="0" VerticalContentAlignment="Stretch"/> <Grid x:Name="PART\_DisabledVisual" Grid.ColumnSpan="2" Grid.Column="0" IsHitTestVisible="False" Opacity="0" Grid.Row="0"> <Grid.ColumnDefinitions> <ColumnDefinition Width="\*"/> <ColumnDefinition Width="Auto"/> </Grid.ColumnDefinitions> <Rectangle Grid.Column="0" Fill="#A5FFFFFF" RadiusX="1" RadiusY="1" Grid.Row="0"/> <Rectangle Grid.Column="1" Fill="#A5FFFFFF" Height="18" Margin="3,0,3,0" RadiusX="1" RadiusY="1" Grid.Row="0" Width="19"/> <Popup x:Name="PART\_Popup" AllowsTransparency="True" Placement="Bottom" PlacementTarget="{Binding ElementName=PART\_TextBox}" StaysOpen="False"/> </Grid> </Grid> </Border> <ControlTemplate.Triggers> <DataTrigger Binding="{Binding Source={x:Static SystemParameters.HighContrast}}" Value="false"> <Setter Property="Foreground" TargetName="PART\_TextBox" Value="{Binding Foreground, RelativeSource={RelativeSource TemplatedParent}}"/> </DataTrigger> </ControlTemplate.Triggers> </ControlTemplate> </Setter.Value> </Setter> </Style> <Style TargetType="DatePickerTextBox"> <Setter Property="IsReadOnly" Value="True"/> </Style> </Application.Resources> </Application>

**Count**

private void btnEdit\_Click(object sender, RoutedEventArgs e) { var rowForRemoving = dgLessons.SelectedItems.Cast<Lessons>().ToList(); if (rowForRemoving.Count == 0) { MessageBox.Show("Выберите урок для редактирования!", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error); } else if (rowForRemoving.Count > 1) { MessageBox.Show("Вы не можете выбрать более одного урока для редактирования!", "Ошибка", MessageBoxButton.OK, MessageBoxImage.Error); } else { Manager.MainFrame.Navigate(new AddEditLessonsPage((Lessons)dgLessons.SelectedItem)); dgLessons.SelectedItems.Clear(); } }