

Министерство образования Московской области

ГАПОУ МО  
«Егорьевский техникум»

Научно-практическая конференция  
по итогам производственной  
(профессиональной) практики на тему:

**“Профессионализм -  
основа успешной карьеры”**



*Сборник докладов и выступлений  
на научно-практической конференции*

г.о. Егорьевск  
08 апреля 2021 года

**План**  
**проведения внеурочного мероприятия научно-практической конференции**  
**по итогам производственной (профессиональной) практики**  
**на тему: «Профессионализм-основа успешной карьеры»**

**Дата проведения:** 08.04.2021

**Ответственные за проведение конференции:**

Зам. директора по УПР *Арбузкина Ольга Викторовна*

Преподаватели специальных дисциплин:

*Степанова Светлана Юрьевна, Шитова Варвара Олеговна*

Руководители производственной практики

*Пир Александр Александрович, Кирилина Ирина Анатольевна*

**Количество участников:** 40 человек

**Участники конференции:**

- представители предприятий (руководители производственной практики от предприятий);
- председатель цикловой методической комиссии преподавателей и мастеров производственного обучения профессионального цикла (общепрофессиональных дисциплин и профессиональных модулей) по специальностям СПО: 09.02.07 Информационные системы и программирование, 10.02.05 Обеспечение информационной безопасности автоматизированных систем; по профессии СПО 09.01.03 Мастер по обработке цифровой информации;
- преподаватели общепрофессиональных и специальных дисциплин, Руководители производственной практики;
- обучающиеся групп Ип-73 и Ип-83 по специальности 09.02.07 Информационные системы и программирование.

**Цель научно-практической конференции:**

- подведение итогов практического обучения студентов, обсуждение проблемных вопросов и нахождение совместного пути их решения;
- проанализировать работу обучающихся, дать объективную информацию об эффективности учебно-производственной практики;
- повышение, закрепление и совершенствование знаний;
- стимулирование интереса обучающихся к будущей производственной деятельности, к своей специальности;
- обобщение и систематизация знаний по общепрофессиональным и специальным дисциплинам;
- поддержка профессионального становления обучающихся;
- характеристика - производственной деятельности обучающихся;
- представление обучающимся 3 курса успехов при прохождении производственной практики.

**Задачи научно-практической конференции:**

- способствовать активному включению обучающихся в процесс самообразования и саморазвития;
- способствовать совершенствованию профессиональных компетенций будущего специалиста;
- оценить совместную работу непосредственных руководителей и обучающихся;
- раскрыть наиболее сложные моменты практики;
- заинтересовать обучающихся в получении углубленных знаний по общепрофессиональным и специальным дисциплинам;
- прививать обучающимся навыки учебно-научной, исследовательской деятельности;
- демонстрация и пропаганда творческого потенциала обучающихся.

**Образовательные цели конференции:**

- ознакомление обучающихся третьего курса с базами практики, возможностями и условиями ее прохождения на данных предприятиях;

- ознакомление обучающихся четвертого курса с возможными местами дальнейшего трудоустройства по избранной специальности;
- систематизация и обобщение знаний обучающихся по специальным и общепрофессиональным дисциплинам: Разработка и эксплуатация удалённых баз данных, Технология разработки программных продуктов, Программное обеспечение компьютерных сетей и др. дисциплины учебного плана;
- демонстрация последних достижений в области высокотехнологичных производств в различных отраслях промышленности нашего города;
- углубление знаний по науке и технике.

#### **Развивающие цели конференции:**

- развитие профессионального интереса к избранной специальности 09.02.07 Информационные системы и программирование;
- формирование навыков коммуникабельности;
- развитие навыков работы с нормативно-технической документацией, с научно-технической литературой;
- овладение знаниями и навыками информационно-коммуникационных технологий.

#### **Воспитательные цели конференции:**

- формирование личности, развитие профессионального мышления, совершенствование самостоятельности обучающихся, воспитание культуры речи;
- воспитание любознательности и инициативности, способствующих развитию творческих способностей.

#### **Научно-исследовательские цели конференции:**

- **создание условий**, способствующих развитию интеллектуального и творческого потенциала студентов и преподавателей, включение их в научно-исследовательскую и проектную деятельность;
- поиск информации в научно-технической документации предприятий, технических библиотеках, через информационно-коммуникационные технологии;
- предоставление возможности развития интеллекта, самостоятельной творческой деятельности с учетом индивидуальных особенностей и способностей.

#### **Используемые средства:**

- тексты докладов; альбомы; рефераты; газеты с новинками современного оборудования; презентации; мультимедийный проектор, компьютер, экран.

#### **Межпредметные связи:**

Разработка программных модулей, Поддержка и тестирование программных модулей, Разработка мобильных приложений, Системное программирование, Технология разработки и защиты баз данных, Язык программирования C#  
Русский язык (грамотная речь, грамотное оформление слайдов презентаций)

#### **Методика подготовительного периода:**

- выбор темы научно-практической конференции и обоснование её актуальности;
- определение конкретных шагов к освещению выбранной темы конференции, т.е. установление количества докладов и формулирование тем докладов, подбор материалов для слайдов;
- распределение докладов между обучающимися группы Ип-73;
- подбор научно-технической литературы для составления докладов и подготовки слайдов;
- оказание методической помощи обучающимся при их работе над составлением докладов и презентаций, проведение консультаций;
- выпуск газет;
- подготовка отчетов по практике;
- подготовка актов зала к проведению мероприятия.

### **Программа проведения научно-практической конференции**

<b>Время проведения</b>	<b>Тема доклада</b>	<b>Докладчики</b>	<b>Место прохождения практики</b>
13.30-13.40	Открытие конференции. Вступительное слово	Зам. директора Техникума по УПР <b>Арбузкина Ольга Викторовна</b>	
13.40-13.55	«Профессия Программист»	Преподаватель специальных дисциплин <b>Степанова Светлана Юрьевна</b>	
13.55-14.10	Язык программирования C#	Преподаватель специальных дисциплин <b>Шитова Варвара Олеговна</b>	
14.10-14.20	Разработка приложения для автоматизации работы сотрудников военкомата	Студент группы Ип-73 <b>Карасев Дмитрий</b>	Служба военно- учетного стола управления по обеспечению безопасности администрации городского округа Егорьевск
14.20-14.30	Разработка с помощью SQL и Visual Studio информационной системы Пенсионный фонд России	Студенты группы Ип-73 <b>Харитонов Иван, Разина Ангелина</b>	Пенсионный фонд России по городскому округу Егорьевск
14.30-14.40	Проектирование программного модуля для автоматизации работы библиотечной системы	Студент группы Ип-73 <b>Шелобанов Иван</b>	Центральная библиотека городского округа Егорьевск
14.40-14.50	Разработка программного модуля «Видеонаблюдение»	Студент группы Ип-73 <b>Гусев Артем</b>	Компания «U- VID.RU», г. Егорьевск
14.50-15.00	Разработка программного обеспечения для автоматизации записи посетителей на спортивные занятия	Студент группы Ип-73 <b>Гладков Илья</b>	Дворец спорта «Егорьевск»
15.00-15.15	Корпоративные сети – один из путей повышения эффективности работы предприятия	Руководитель производственной практики в группе Ип- 73, ведущий программист ГАПОУ МО «Егорьевский техникум» <b>Пир Александр Александрович</b>	
15.15-15.25	Автоматизированные информационные системы	Преподаватель специальных дисциплин <b>Кирилина Ирина Анатольевна</b>	

15.25-15.35	Вручение благодарственных писем за участие в конференции	Зам. директора Техникума по УПР <b>Арбузкина Ольга Викторовна</b>	
15.35-15.40	Решение научно-практической конференции	Руководитель производственной практики в группе Ип-73, ведущий программист ГАПОУ МО «Егорьевский техникум» <b>Пир Александр Александрович</b>	
15.40-16.00	Подведение итогов. Закрытие конференции	Зам. директора Техникума по УПР <b>Арбузкина Ольга Викторовна</b>	

Открыла научно-практическую конференцию заместитель директора по учебно-производственной работе - **Арбузкина Ольга Викторовна**.

Здравствуйтесь дорогие участники и гости конференции!

В сегодняшней действительности самыми востребованными на рынке труда специалистами в России и за рубежом являются специалисты в области информационных технологий. Нет сомнений в том, что в наш век информационных технологий наибольшее преимущество получает не тот, кто умеет просто пользоваться компьютером, применяя уже готовые программы, а тот, кто способен совершенствовать старые программы и создавать новые.

Программист – одна из самых перспективных профессий не только в настоящем, но и обозримом будущем.

Современный прогресс требует от информационных технологий постоянного совершенствования и развития. Непрерывно возникают задачи, требующие быстрого решения средствами стандартных компьютерных программ, таких как электронные таблицы, базы данных и др.

Учебная и производственная практика является составной частью учебно-воспитательного процесса и имеет важное значение в подготовке квалифицированного специалиста. Она направлена на закрепление и углубление знаний и умений, полученных студентами в процессе обучения, а также овладением системой профессиональных умений и навыков и первоначальным опытом.

Сегодня мы проводим научно-практическую конференцию по итогам производственной практики по ПМ-02 Осуществление интеграции программных модулей на тему «Профессионализм-основа успешной карьеры». Студенты группы Ип-73 являются выпускниками 2021 года. Закончен этап производственной практики по профилю специальности, которая проходила на предприятиях нашего города, таких как: Егорьевск-обувь, ООО «Стилист», ЕТИ ФГБОУ ВПО МГТУ «СТАНКИН», Автоколонна №1796 филиал, ООО «Бытпласт», ЗАО «Геден Рихтер – РУС», Управление социальной защиты, ООО «СтройТехГрупп», ООО «Мосэнергосбыт» и других.



**Арбузкина Ольга Викторовна –**  
заместитель директора по учебно-производственной работе

Целью производственной практики является изучение студентами реального предприятия и условий работы на нем, получение прикладных навыков в разработке и сопровождении программ, изучении информационных потоков и документооборота, способов хранения и обработки информации, сбор материалов для отчета и предварительный выбор вероятной темы дипломного проекта.

С 20 апреля в группе Ип-73 начинается преддипломная практика. Преддипломная практика студентов является завершающим этапом обучения. Основной целью производственной (преддипломной) практики является сбор материалов для дипломного проектирования, практическая работа совместно с разработчиками профессионалами по созданию программных продуктов, которые будут являться одной из основных частей завершённой выпускной квалификационной работы (дипломной работы).

А сегодня мы послушаем доклады наших ведущих преподавателей и доклады студентов об итогах производственной практики, которые они нам подготовили.

Первое слово я предоставляю преподавателю общепрофессиональных дисциплин **Степановой С.Ю.**

## Доклад на тему: **«Профессия Программист»**

### Программист

Информационные технологии развиваются очень быстро, поэтому для программистов имеет первостепенное значение владение навыками самообразования, чтобы после получения основного диплома идти в ногу со временем и не терять квалификации. Практически любая компания сегодня нуждается в автоматизации своего бизнеса, и спрос на таких специалистов намного превышает предложение, следовательно, даже у вчерашнего выпускника есть все шансы найти хорошую работу.

#### Общее описание

Программист – это специалист, занимающийся разработкой программного обеспечения (ПО) для персональных, встраиваемых, промышленных и других разновидностей компьютеров, то есть программированием. Это может быть как руководитель крупных проектов по разработке ПО, так и «одиночка», пишущий код конкретной программы. Перед большинством предприятий и организаций рано или поздно встает вопрос автоматизации, и многие управленцы стремятся использовать достижения современных информационных технологий, поэтому программист является своего рода консультантом, выполняющим посредническую функцию между тем, чего желает получить руководитель, и тем, что предлагает на данный момент мир высоких технологий.

Различают системных и прикладных программистов. Прикладной – это специалист, который осуществляет разработку и отладку программ для решения различных задач. Системный – специалист, занимающийся разработкой, эксплуатацией и сопровождением системного программного обеспечения.

#### Образование

Программистов готовят во множестве высших учебных заведений. Одни из самых авторитетных в столице – это МГУ им. М. В. Ломоносова (механико-математический факультет и факультет вычислительной математики и кибернетики), МФТИ, МИФИ, МГТУ им. Н. Э. Баумана, МИРЭА, МИЭМ, МЭСИ, МАИ. По данным компании RAND, ежегодно российские вузы выпускают до 100 тысяч дипломированных специалистов в области программирования.



**Степанова Светлана Юрьевна –  
преподаватель общепрофессиональных  
дисциплин**

Самым важным в процессе обучения является не столько прослушивание лекций, сколько решение задач и участие в различных учебных проектах. Будущему программисту важно научиться думать, находить верные решения, «шевелить мозгами». Информационные технологии развиваются очень быстро, поэтому для таких специалистов необходимы навыки самообразования, чтобы после получения основного диплома идти в ногу со временем и не терять квалификации.

### **Смежные карьеры**

В отличие от своих зарубежных коллег многие российские программисты являются по образованию физиками, химиками или математиками. И, как показывает практика, из таких выпускников получаются первоклассные ИТ-специалисты. Известно, что программисты из Индии, Китая, Вьетнама, Индонезии широко используются там, где требуется массовое индустриальное программирование, но в тех разработках, где нужен прежде всего поиск творческих решений, российские профессионалы считаются наиболее сильными во всем мире. В какой-то степени это обусловлено тем, что наши соотечественники получают качественное фундаментальное образование.

На факультете высшей математики и кибернетики МГУ разработана система дополнительного двухлетнего вечернего обучения, позволяющая готовить программистов из обладателей дипломов по смежным специальностям и направлениям. Этот проект получил название «Высшая компьютерная школа "Эксперт"». Выпускники таких программ получают государственный диплом о дополнительной квалификации («разработчик компьютерных технологий») и несколько международных сертификатов в зависимости от направления специализации. В ближайшем будущем профессорско-преподавательский состав МГУ планирует осуществлять подготовку, используя технологии дистанционного обучения.

### **Функциональные обязанности программиста**

Разрабатывать новые операционные системы, приложения и программы наподобие MS Office или Adobe Photoshop, работая в России, программисту вряд ли придется. Этим занимаются профессионалы высшей квалификации таких иностранных фирм-производителей, как Microsoft, Sun, Oracle, Adobe и других. В отечественных компаниях подобные проекты – редкость, и пользователи, как правило, устанавливают на свои ПК в основном западные продукты. Отдельные российские фирмы-разработчики программного обеспечения существуют, но их немного, например «Лаборатория Касперского», которая выпускает антивирусные программы. Созданием программ занимаются и государственные структуры: НИИ и конструкторские бюро военно-промышленного комплекса, космонавтики, но их крайне мало.

В основном в России требуются программисты-"прикладники": практически любая компания сегодня нуждается в автоматизации своего бизнеса, в возможности работать по внутренней компьютерной сети и быстро обмениваться информацией в электронном виде между подразделениями и отделами, вести учет продукции, закупок, реализованного товара. Они адаптируют, а если нужно, и разрабатывают пакет специальных программ, учитывая особенности бизнеса организации: одно дело автоматизировать деятельность издательского дома, другое – торговой фирмы или ресторана. Однако многим предприятиям требуется не только автоматизация бизнеса, но и постоянный мониторинг налаженной системы, ее изменение в случае расширения компании, поэтому работодатели готовы к зачислению программистов к себе в штат. Все же некоторые организации предпочитают заказывать автоматизацию своего бизнеса специализированным ИТ-компаниям, предоставляющим такие услуги.

Особым спросом на рынке труда пользуются программисты 1С – те же «прикладники», но адаптирующие уже готовый пакет программ фирмы «1С» к нуждам конкретного предприятия, главным образом бухгалтерии. Анализируя требования заказчика, они устанавливают программы, которые облегчают расчеты, оптимизируют работу персонала, налаживают документооборот.

Еще одна сфера деятельности программиста – web-программирование: разработка, модернизация и поддержка сайтов. Эта область требует многих специфических знаний – PHP, MYSQL, XHTML, CSS, JavaScript, XML. Причем очень часто работодатели желают получить профессионала «два в одном» – web-дизайнера и web-программиста, чтобы он не только решал технические вопросы (быстрая загрузка сайта, работа ссылок, поисковика), но также занимался и оформлением сайта, а иногда даже и его информационным наполнением.

### **Профессиональные навыки**

Практически ежемесячно выходят новые версии программ, обновляются характеристики оборудования, и специалисты в области информационных технологий должны быть всегда в курсе этих изменений. Поэтому способность к самообучению – один из главных навыков, которым должен обладать программист.

Владение английским языком на уровне чтения технической документации является еще одним обязательным требованием, предъявляемым к представителям этой профессии. Для таких специалистов очень важно умение работать: в команде, над большими проектами, со средствами коллективной разработки, с крупными финансовыми системами (бюджетными, банковскими, управленческого учета).

Для претендентов на позицию ведущего программиста желательны навыки управления проектами и коллективом, самостоятельность, инициативность, а также способность нести личную ответственность за поставленную задачу.

### **Плюсы и минусы**

К плюсам профессии можно отнести все возрастающую востребованность ее представителей. Сфера информационных технологий в России стремительно развивается, и темпы ее роста, составляя около 25% в год, намного обгоняют аналогичные показатели в Европе и США. К тому же спрос на специалистов постоянно превышает предложение, так что даже начинающему программисту не составляет особого труда найти работу. Еще одно преимущество профессии программиста – возможность работать не только в нашей стране, но и за рубежом, где российские ИТ-специалисты ценятся очень высоко.

Но все же, несмотря на утверждение профессионалов, что программирование сродни творчеству, труд программиста требует высокой концентрации внимания и больших трудозатрат. К минусам профессии можно отнести и так называемый комплекс «козла отпущения». Например, если вы устроились на работу в компанию, занимающуюся непосредственным производством ПО, то имейте в виду, что число фирм, способных выступить как заказчики, невелико, поэтому менеджеры пытаются ухватить любой «денежный» заказ, не особо задумываясь о том, какими средствами и в какие сроки он может быть выполнен. А расплачиваться за подобное «планирование» приходится обычно программистам-разработчикам посредством вечерних и ночных «бдений» в офисе и работы в выходные дни. Тем не менее, различные минусы профессии во многом уравниваются немалым размером заработной платы, хорошими карьерными перспективами и возможностью реализации своих творческих замыслов.

### **Оплата труда программиста**

По данным статистики компании HeadHunter, минимальная зарплата, на которую может рассчитывать начинающий программист, начинается от \$400. В большинстве же столичных компаний разработчикам программного обеспечения предлагается вознаграждение в размере от \$1000 до \$3500. Как сообщается в докладе, опубликованном в журнале Money и на сайте Salary.com, размеры компенсации программистов в ближайшее время будут только увеличиваться.

### **Перспективы**

Талантливый программист в перспективе может стать руководителем крупного проекта по разработке программного обеспечения. Однако существует мнение, что из них получаются не очень хорошие менеджеры. Как правило, такие специалисты зарабатывают очень хорошо, поэтому к продвижению по карьерной лестнице стремятся далеко не многие. В большинстве случаев целью личного развития программиста становится повышение профессионального уровня и реализация творческих задумок. Яркий пример – Алексей Пажитнов, создатель оригинальной авторской программы «Тетрис». Раньше он работал в вычислительном центре Российской академии наук, а сейчас – в Microsoft, и все, что он делает, пользуется спросом. Его программы продаются под названием «Игры от Пажитнова» – это уже бренд!

Доклад на тему:

**«Язык программирования C#»**



## Язык программирования C#

Язык программирования C# был разработан в 1993—2001 годах группой инженеров компании Microsoft под руководством Андерса Хейлсберга и Скотта Вильтаумота как язык разработки приложений для платформы Microsoft.

К 2000 году у Microsoft были готовы промышленные версии новых технологий и решений для обмена сообщениями и данными, а также для создания Internet-приложений. Была выпущена и новая платформа для разработки под новые решения — .NET. В ней объединились сразу несколько языков программирования, что было в новинку для того времени.

Ещё одним новшеством платформы .NET была технология активных серверных страниц ASP.NET (Active Server Page). С её помощью можно было относительно быстро разработать веб-приложения, взаимодействующие с базами данных. Специально для ASP.NET был создан язык программирования C#. Да и сама ASP.NET была полностью написана на нём.

C# — это язык с C-подобным синтаксисом. Здесь он близок в этом отношении к C++ и Java.

Будучи объектно-ориентированным языком, он много перенял у Java и C++. Как и Java, C# изначально предназначался для веб-разработки, и примерно 75% его синтаксических возможностей такие же, как у Java. C# также называют «очищенной версией Java». Ещё 10% наш герой позаимствовал из C++ и 5% — из Visual Basic. Оставшиеся 10% C# — это реализация собственных идей разработчиков. Объектно-ориентированный подход позволяет строить с помощью C# крупные, но в то же время гибкие, масштабируемые и расширяемые приложения.

C# уже давно поддерживает много полезных функций:

- инкапсуляция,
- наследование,
- полиморфизм,
- перегрузка операторов,
- статическая типизация.

При этом он всё ещё активно развивается, и с каждой новой версией появляется всё больше интересного — например лямбды, динамическое связывание, асинхронные методы и т.д.

По сравнению с другими языками C# довольно молод, но в то же время он уже прошёл большой путь. Первая версия языка вышла вместе с релизом Microsoft Visual Studio .NET в феврале 2002 года. Текущей версией языка является версия C# 8.0, которая вышла в сентябре 2019 года вместе с релизом .NET Core 3.

Когда говорят C#, нередко имеют в виду технологии платформы .NET (Windows Forms, WPF, ASP.NET, Xamarin). И наоборот, когда говорят .NET, нередко имеют в виду C#. Однако, хотя эти понятия связаны, отождествлять их неверно. Язык C# был создан специально для работы с фреймворком .NET, однако само понятие .NET несколько шире.

Когда-то Билл Гейтс сказал, что .NET — это лучшее, что сделала компания Microsoft. У него есть весомые основания так считать. Фреймворк .NET представляет мощную платформу для создания приложений. Вот его киллер-фичи:

- Поддержка нескольких языков. В основе .NET — общезыковая среда исполнения Common Language Runtime (CLR), благодаря чему платформа поддерживает несколько языков: наряду с C# это VB.NET, C++, F#, а также различные диалекты других языков, привязанные к .NET, например, Delphi.NET. Код на любом из этих языков компилируется в сборку на общем



**Шитова Варвара Олеговна –**  
преподаватель общепрофессиональных  
дисциплин

языке CIL (Common Intermediate Language) — своего рода ассемблер платформы .NET. Поэтому можно сделать отдельные модули одного приложения на разных языках.

- Мощная библиотека классов. .NET представляет единую для всех поддерживаемых языков библиотеку классов. Какое бы приложение мы ни собирались писать на C# — текстовый редактор, чат или сложный веб-сайт — так или иначе мы задействуем библиотеку классов .NET.
- Разнообразие технологий. Общеязыковая среда исполнения CLR и базовая библиотека классов — это основа для целого стека технологий, которые разработчики могут задействовать при создании разных приложений. Например, для баз данных в этом стеке имеется технология ADO.NET и Entity Framework Core. Для графических приложений с насыщенным интерфейсом — технологии WPF и UWP. Для более простых графических приложений — Windows Forms. Для разработки мобильных приложений — Xamarin. Для создания веб-сайтов - ASP.NET и т.д.

.NET долгое время развивался под названием .NET Framework — преимущественно как платформа для Windows. Но с 2019 она больше не развивается — последней версией этой платформы стала .NET Framework 4.8.

В 2014 Microsoft начал выпускать альтернативную платформу - .NET Core, которая должна была вобрать в себя все возможности устаревшего .NET Framework и добавить новую функциональность. Поэтому следует различать .NET Framework, который предназначен преимущественно для Windows, и кроссплатформенный .NET Core.

У «шарпа» выделяют много преимуществ:

- Поддержка подавляющего большинства продуктов Microsoft
- Бесплатность ряда инструментов для небольших компаний и некоторых индивидуальных разработчиков — Visual Studio, облако Azure, Windows Server, Parallels Desktop для Mac Pro и др.
- Типы данных имеют фиксированный размер (32-битный int и 64-битный long), что повышает «мобильность» языка и упрощает программирование, так как вы всегда знаете точно, с чем вы имеете дело.
- Автоматическая «сборка мусора» Это значит, что нам в большинстве случаев не придётся заботиться об освобождении памяти. Вышеупомянутая общеязыковая среда CLR сама вызовет сборщик мусора и очистит память.
- Большое количество «синтаксического «сахара» — специальных конструкций, разработанных для понимания и написания кода. Они не имеют значения при компиляции.
- Низкий порог вхождения. Синтаксис C# имеет много схожего с другими языками программирования, благодаря чему облегчается переход для программистов. Язык C# часто признают наиболее понятным и подходящим для новичков.
- С помощью Xamarin на C# можно писать программы и приложения для таких операционных систем, как iOS, Android, MacOS и Linux;
- Сегодня в любом регионе России имеется немало вакантных мест на должность C#-программиста.

Но есть у C# и некоторые недостатки:

- Приоритетная ориентированность на платформу Windows;
- Язык бесплатен только для небольших фирм, индивидуальных программистов, стартапов и учащихся. Крупной компании покупка лицензионной версии этого языка обойдётся в круглую сумму.

C# не представляет сложности для новичков, так как его сравнительно легко изучить и понять. На просторах интернета можно найти множество курсов и онлайн-школ по обучению C#, предлагающих обучение продолжительностью от 1 до 6 месяцев. Также существуют экспресс-курсы для «чайников», где новичков обучают основам за пару дней. Вообще, C# намного проще будет освоить, если вы уже знаете C, C++ или Java.

C# на протяжении долгого времени уверенно держит позиции в рейтинге наиболее востребованных на рынке разработки языков. Сначала им интересовались только разработчики под Windows, но затем C# научился работать на Mac OS, Linux, iOS и Android. А после того как код платформы открыли для всех желающих, были сняты практически все возможные ограничения в применении C#. В результате язык активно развивается и применяется всё шире. Его часто рекомендуют к изучению в качестве одного из базовых для разработчиков любого профиля.

Инструментарий С# позволяет решать широкий круг задач, язык действительно очень мощный и универсальный. На нём часто разрабатывают:

- веб-приложения,
- игры,
- мобильные приложения для Android или iOS,
- программы под Windows.

Перечень возможностей разработки практически не имеет ограничений благодаря широчайшему набору инструментов и средств. Конечно, всё это можно реализовать при помощи других языков. Но некоторые из них узкоспециализированные, а в некоторых придётся использовать дополнительные инструменты сторонних разработчиков. В С# решить широкий круг задач возможно быстрее, проще и с меньшими затратами времени и ресурсов.

Что касается зарплаты программистов на С#, то по [статистике](#) Хабр.Карьеры она обычно варьируется в диапазоне от 60 до 200 тысяч рублей и в среднем составляет 100 тысяч рублей .

Доклад на тему:

### ***«Разработка приложения для автоматизации работы сотрудников военкомата»***

**Место прохождения практики:**  
Служба военно-учетного стола управления по обеспечению безопасности администрации городского округа Егорьевск

Военкомат представляет собой орган местного самоуправления. Он отвечает за учетно-призывную и военно-мобилизационную деятельность. Полное название – военный комиссариат.

Задачи и функции Военкомата  
Военкоматы и военные комиссариаты (ВК) занимаются учетом военнослужащих в запасе и

призывников, организуют медосмотры, на которых делают выводы о состоянии призывников. В военное время работа в военкоматах будет вестись в усиленном режиме.

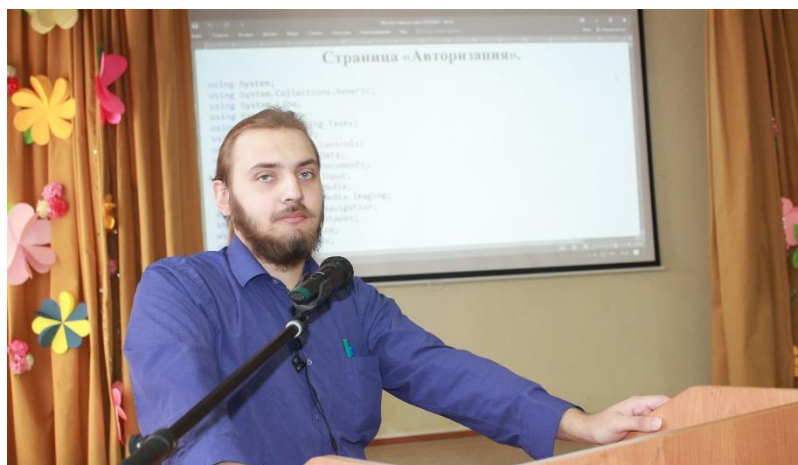
Военнообязанных граждан распределяют по специальностям, осуществляют комиссию для военнослужащих, содержащихся в запасе.

Военный комиссариат занимается исполнением таких задач, как:

- Учитывать людской ресурс.
- Организовывать подготовку молодых граждан к несению воинской службы.
- Проводить призывы для прохождения военной службы.
- Обеспечивать возможность приема на службу по контракту.
- Осуществлять прочие оборонные мероприятия.
- Работать с обеспечением пенсии для служащих, находящихся в запасе.

Также в военкомате производится учет граждан, которые являются ветеранами и находятся в запасе. Организация занимается проведением информирующих мероприятий для населения, принимает участие в розыске погибших воинов, работает с документами военных архивов.

На производственной практике я занимался разработкой приложения для автоматизации работы сотрудников военкомата.



**Карасев Дмитрий –**  
студент группы Ип-73

### **Visual Studio 2019**

Интегрированная среда разработки Visual Studio — это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений.

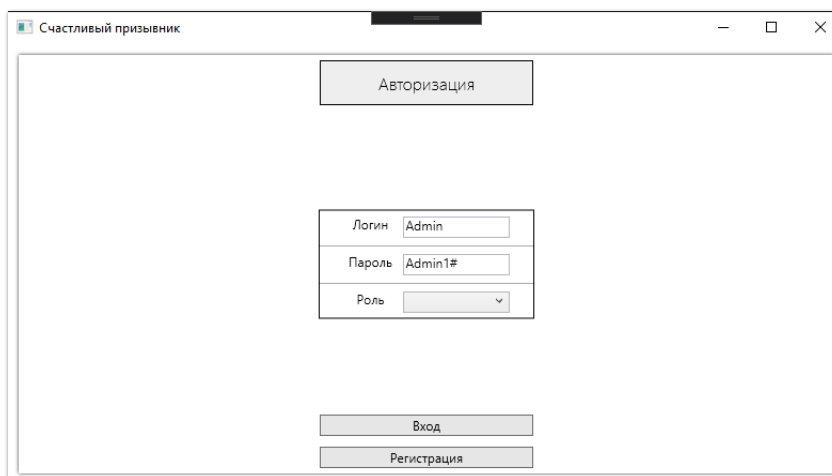
Интегрированная среда разработки (IDE) представляет собой многофункциональную программу, которую можно использовать для различных аспектов разработки программного обеспечения.

Помимо стандартного редактора и отладчика, которые существуют в большинстве сред IDE, Visual Studio включает в себя компиляторы, средства автозавершения кода, графические конструкторы и многие другие функции для упрощения процесса разработки.

### Предметная область

- ◆ Приложение разрабатывается как прототип. В нем будет присутствовать возможность для расширения.
- ◆ Приложение разрабатывается как для работы в нем работников военкомата, так и для работы в нем обычных пользователей.
- ◆ В приложении должно быть предусмотрено базовые функции для работы в нем, а именно: отображение, фильтрация, удаление, редактирование данных, все функции должны быть разбиты по ролям.

### Страница «Авторизация»



Счастливый призzyвик

Авторизация

Логин: Admin

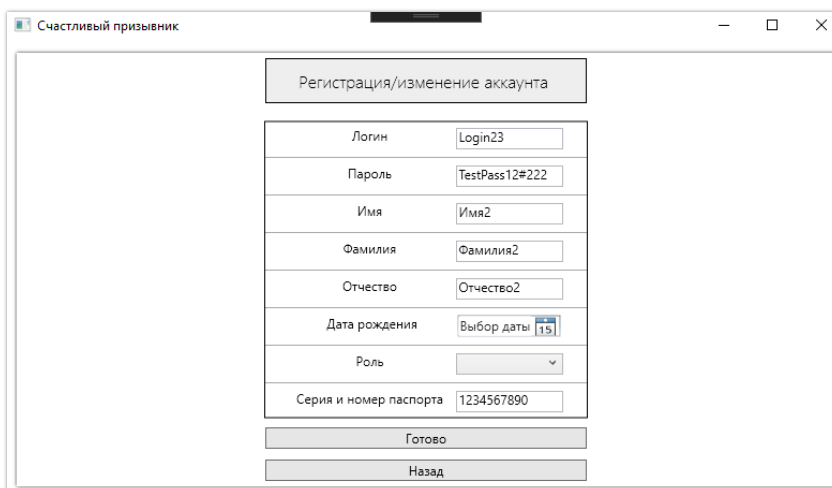
Пароль: Admin1#

Роль: [v]

Вход

Регистрация

### Страница «Регистрация/изменение аккаунта». При переходе с кнопки «Регистрация»



Счастливый призzyвик

Регистрация/изменение аккаунта

Логин: Login23

Пароль: TestPass12#222

Имя: Имя2

Фамилия: Фамилия2

Отчество: Отчество2

Дата рождения: Выбор даты 15

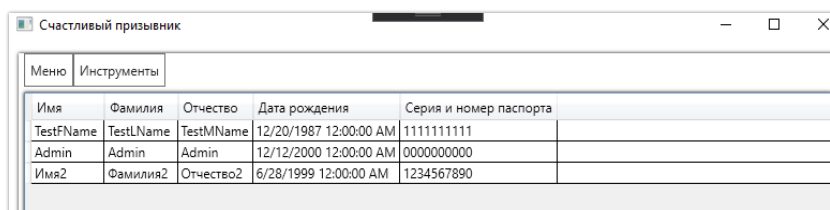
Роль: [v]

Серия и номер паспорта: 1234567890

Готово

Назад

### Главная страница «Информация»



Счастливый призzyвик

Меню | Инструменты

Имя	Фамилия	Отчество	Дата рождения	Серия и номер паспорта
TestFName	TestLName	TestMName	12/20/1987 12:00:00 AM	1111111111
Admin	Admin	Admin	12/12/2000 12:00:00 AM	0000000000
Имя2	Фамилия2	Отчество2	6/28/1999 12:00:00 AM	1234567890

### Окно «Поиск»

Поиск

Фамилия

Имя

Отчество

Дата рождения

Выбор даты 15

Серия и номер паспорта

Найти

Отмена

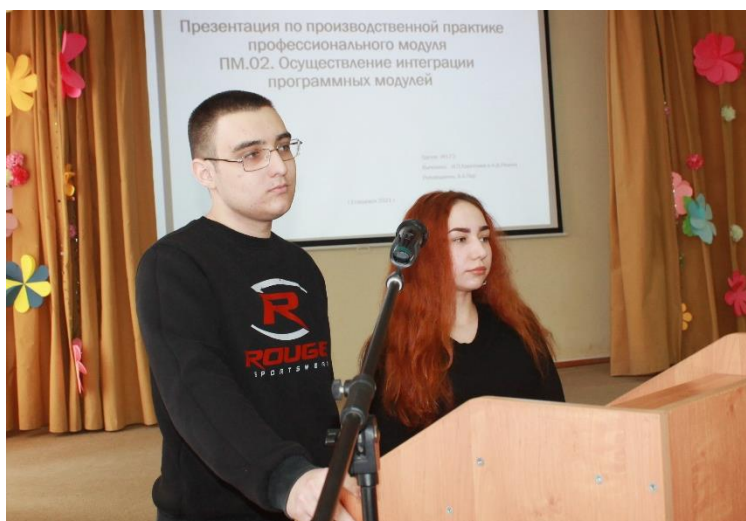
Доклад на тему:

## **«Разработка с помощью SQL и Visual Studio информационной системы «Пенсионный фонд России»»**

**Место прохождения практики:**  
*Пенсионный фонд РФ по городскому округу Егорьевск*

В этой работе представлен проект разработанный за все время, проведенное на практике. Это информационная система ПФР - государственное учреждение, занимающееся обязательным социальным обеспечением, разработанная с помощью SQL и Visual Studio.

Информационная система ПФР – разработанная для сотрудников пенсионного фонда РФ. Для удобного взаимодействия с данными и подсчетом коэффициента и любых чисел в целом а также для быстрого просмотра статистики ПФР.



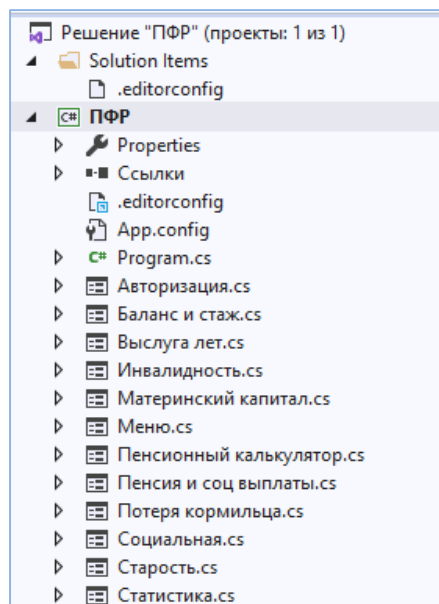
**Харитонов Иван и Разина Ангелина – студенты группы Ип-73**

### **1.Разработка алгоритма решения поставленной задачи**



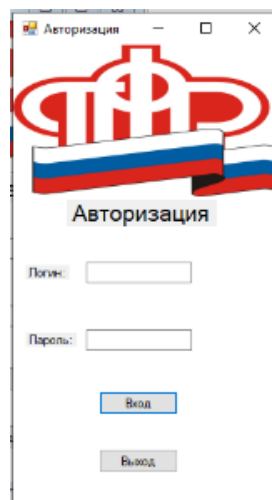
## 2.Проектирование программного модуля

Формы проекта:

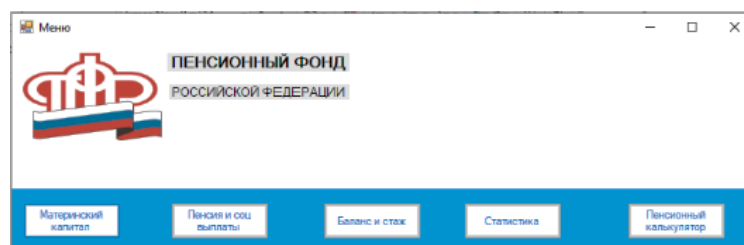


Формы содержат все необходимые элементы:

### Авторизация



### Меню



## Список: Материнский капитал

Ф.И.О.	Дата рождения	Пол	Место рождения	Гражданство	Коэффициент выплат	Сумма	Определяется до	Ф.И.О. Опекун
Колетов Анатолий Андреевич	2017-08-17	М	г. Егорьевск	РФ	10,7	450000.0000	2022-08-17	Колетова Анна Сергеевна
Ружина Елизавета Юрьевна	2019-06-15	Ж	г. Москва	РФ	12	450000.0000	2025-06-15	Ружина Валентина Степановна
Крюков Александр Андреевич	2016-01-04	М	г. Воронеж	РФ	8	450000.0000	2022-01-04	Крюкова Людмила Геннадьевна
Ларин Антон Альбертович	2020-05-09	М	г. Егорьевск	РФ	10,7	450000.0000	2027-05-09	Ларина Мария Михайловна
Кропалькина Антонина Ивановна	2019-05-05	Ж	г. Москва	РФ	12	450000.0000	2026-05-05	Кропалькина Надежда Геннадьевна
Лобина Вера Васильевна	2016-03-08	Ж	г. Егорьевск	РФ	10,7	450000.0000	2022-03-08	Лобина Мария Александровна
Рыбин Артем Иванович	2016-05-08	М	г. Егорьевск	РФ	10,7	450000.0000	2022-05-08	Рыбина Александра Ивановна
Сидоров Иван Иванович	2016-08-12	М	г. Егорьевск	РФ	10,7	450000.0000	2022-08-12	Сидорова Валентина Ивановна
Куркин Максим Александрович	2020-05-08	М	г. Москва	РФ	12	450000.0000	2027-05-08	Куркина Екатерина Егоровна
Темченко Анна Борисовна	2020-05-08	Ж	г. Москва	РФ	12	450000.0000	2027-05-08	Темченко Раиса Зиновьевна

Сохранить    Назад

## Окно: Пенсии и соц. выплаты

Пенсия и ...

Виды пенсий

- Выслуга лет
- Старость
- Инвалидность
- Потеря кормильца
- Социальная

Назад

калькулятор

Пенсионный калькулятор

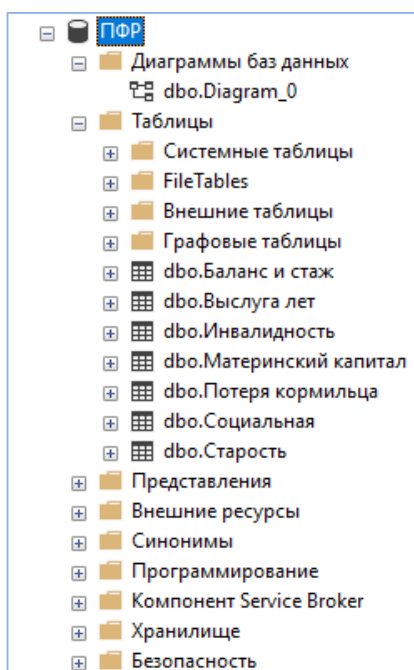
%    √    x<sup>2</sup>    1/x

Остальные 7 форм: пенсия и соц. выплаты, баланс и стаж, статистика, выслуга лет, старость, инвалидность, потеря кормильца и социальная аналогичны

### 3. Программирование программного модуля

В начале была создана БД в SQL, таблицы и диаграмма. После создания БД, БД была привязана к Visual Studio.

База данных:



База данных в SQL:

Таблицы:

Имя столбца	Тип данных	Разрешить ...
[Ф.И.О.]	nchar(50)	<input checked="" type="checkbox"/>
[Дата рождения]	date	<input checked="" type="checkbox"/>
Гражданство	nchar(10)	<input checked="" type="checkbox"/>
Баланс	money	<input checked="" type="checkbox"/>
Стаж	nchar(10)	<input checked="" type="checkbox"/>
Адрес	nchar(50)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>



Таблица (проект)

Ф.И.О.	Дата рождения	Гражданство	Баланс	Стаж	Адрес
Крыанов Иван Иванович	1978-07-09	РФ	1500000,0000	15	г.Егорьевск 5-6-234
Кулаков Игорь Игоревич	1987-02-07	РФ	1500000,0000	34	г.Егорьевск 7-3-124
Сарокина Ирина Валерьевна	1998-02-06	РФ	1500000,0000	5	г.Егорьевск 2-9-154
Сидоров Анатолий Иванович	1996-10-11	РФ	1500000,0000	10	г.Егорьевск 5-40-22
Рубильников Игорь Игоревич	1988-06-01	РФ	1500000,0000	12	г.Егорьевск 7-4-211
Пирогова Людмила Геннадевна	1978-09-08	РФ	1500000,0000	15	г.Егорьевск 3-5-155
Сырысин Эдуард Андреевич	1967-03-06	РФ	1500000,0000	10	г.Егорьевск 1-15-23
Рульков Иван Иванович	1999-02-05	РФ	1500000,0000	3	г.Егорьевск 2-5-32
Икоткин Игорь Валерьевич	1993-10-10	РФ	1500000,0000	10	г.Егорьевск 5-6-21
Петров Петр Петрович	1991-11-11	РФ	1500000,0000	10	г.Егорьевск 5-5-167
** NULL	NULL	NULL	NULL	NULL	NULL

Таблица (список)

Имя столбца	Тип данных	Разрешить ...
[Ф.И.О.]	nchar(50)	<input checked="" type="checkbox"/>
[Дата рождения]	date	<input checked="" type="checkbox"/>
[Место рождения]	nchar(50)	<input checked="" type="checkbox"/>
Гражданство	nchar(50)	<input checked="" type="checkbox"/>
Пол	nchar(10)	<input checked="" type="checkbox"/>
Сумма	money	<input checked="" type="checkbox"/>
[Кoeffициент выплат]	nchar(10)	<input checked="" type="checkbox"/>
Адрес	nchar(50)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Таблица (проект)

Ф.И.О.	Дата рождения	Место рожде...	Гражданство	Пол	Сумма	Кoeffициент...	Адрес
Копустин Дмитрий Васильевич	1986-02-09	г.Егорьевск	РФ	М	25000,0000	12	г.Егорьевск 6-5-12
Полов Иван Павлович	1984-11-02	г.Егорьевск	РФ	М	25000,0000	12	г.Егорьевск 7-23-122
Никитин Максим Генадьевич	1976-12-04	г.Егорьевск	РФ	М	25000,0000	12	г.Егорьевск 8-43-12
Икоткин Петр Петрович	1982-12-07	г.Егорьевск	РФ	М	25000,0000	12	г.Егорьевск 9-2-165
Гончарова Ильвира Степановна	1973-05-05	г.Егорьевск	РФ	Ж	25000,0000	12	г.Егорьевск 6-45-12
Копытин Игнат Сергеевич	1987-02-06	г.Егорьевск	РФ	М	25000,0000	12	г.Егорьевск 7-9-76
Кислова Мария Михайловна	1987-02-02	г.Егорьевск	РФ	Ж	25000,0000	12	г.Егорьевск 9-87-97
Терещина Екатерина Паловна	1988-09-05	г.Егорьевск	РФ	Ж	25000,0000	12	г.Егорьевск 7-8-245
Жирнов Александр Васильевич	1974-12-09	г.Егорьевск	РФ	М	25000,0000	12	г.Егорьевск 5-23-153
Рикова Анастасия Максимовна	1965-02-04	г.Егорьевск	РФ	Ж	25000,0000	12	г.Егорьевск 7-22-34

Таблица (список)

Имя столбца	Тип данных	Разрешить ...
[Ф.И.О.]	nchar(50)	<input checked="" type="checkbox"/>
[Дата рождения]	date	<input checked="" type="checkbox"/>
[Место рождения]	nchar(50)	<input checked="" type="checkbox"/>
Гражданство	nchar(50)	<input checked="" type="checkbox"/>
Пол	nchar(10)	<input checked="" type="checkbox"/>
Сумма	money	<input checked="" type="checkbox"/>
[Кoeffициент выплат]	nchar(10)	<input checked="" type="checkbox"/>
Адрес	nchar(50)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Таблица (проект)

Ф.И.О.	Дата рождения	Место рожде...	Гражданство	Пол	Сумма	Коэффициент...	Адрес
Икутин Игорь Анатольевич	1978-07-09	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 5-12-218
Ярикова Авдосья Андреевна	1987-12-12	г.Егорьевск	РФ	Ж	12000,0000	7,2	г.Егорьевск 1-2-176
Карпова Ирина Степановна	1976-11-09	г.Егорьевск	РФ	Ж	12000,0000	7,2	г.Егорьевск 4-1-156
Кулысов Алексей Владимирович	1984-12-11	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 2-12-09
Кирков Альберт Геннадьевич	1987-03-08	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 1-5-231
Пронин Максим Федотович	1992-09-02	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 6-231-145
Аникин Владислав Антонович	1999-12-09	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 7-233-134
Овцова Марина	1978-04-09	г.Егорьевск	РФ	Ж	12000,0000	7,2	г.Егорьевск 5-12-321
Коровин Максим Иванович	1965-03-09	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 2-127-32
Баранов Иван Петрович	1987-07-01	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 7-26-65

Таблица (список)

Имя столбца	Тип данных	Разрешить ...
[Ф.И.О.]	nchar(50)	<input checked="" type="checkbox"/>
[Дата рождения]	date	<input checked="" type="checkbox"/>
Пол	nchar(10)	<input checked="" type="checkbox"/>
[Место рождения]	nchar(50)	<input checked="" type="checkbox"/>
Гражданство	nchar(50)	<input checked="" type="checkbox"/>
[Коэффициент выплат]	nchar(10)	<input checked="" type="checkbox"/>
Сумма	money	<input checked="" type="checkbox"/>
[Определиться до]	date	<input checked="" type="checkbox"/>
[Ф.И.О. Опекунa]	nchar(50)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Таблица (проект)

Ф.И.О.	Дата рождения	Пол	Место рожде...	Гражданство	Коэффициент ...	Сумма	Определиться...	Ф.И.О. Опекунa
Котлетов Анатолий Андреевич	2017-08-17	М	г.Егорьевск	РФ	10,7	450000,0000	2023-08-17	Котлетова Анна Сергеевна
Ручкина Елизавета Юриевна	2019-06-15	Ж	г.Москва	РФ	12	450000,0000	2025-06-15	Ручкина Валентина Степановна
Крюкова Александр Андреевич	2016-01-04	М	г.Воронеж	РФ	8	450000,0000	2022-01-04	Крюкова Людмила Геннадьевна
Ларин Антон Албертович	2020-05-09	М	г.Егорьевск	РФ	10,7	450000,0000	2027-05-09	Ларина Мария Михайловна
Куропаткина Антонина Ивановна	2019-05-05	Ж	г.Москва	РФ	12	450000,0000	2026-05-05	Куропаткина Надежда Геннадьевна
Лобина Вера Васильевна	2016-03-08	Ж	г.Егорьевск	РФ	10,7	450000,0000	2022-03-08	Лобина Мария Александровна
Рыбин Артем Иванович	2016-05-08	М	г.Егорьевск	РФ	10,7	450000,0000	2022-05-08	Рыбина Алефтина Ивановна
Сидоров Иван Иванович	2016-08-12	М	г.Егорьевск	РФ	10,7	450000,0000	2022-08-12	Сидорова Валентина Ивановна
Кулрин Максим Александрович	2020-05-08	М	г.Москва	РФ	12	450000,0000	2027-05-08	Кулрина Екатерина Егоровна
Темцино Анна Борисовна	2020-05-08	Ж	г.Москва	РФ	12	450000,0000	2027-05-08	Темцино Раиса Зиновьевна

Таблица (список)

Имя столбца	Тип данных	Разрешить ...
[Ф.И.О.]	nchar(50)	<input checked="" type="checkbox"/>
[Дата рождения]	date	<input checked="" type="checkbox"/>
[Место рождения]	nchar(50)	<input checked="" type="checkbox"/>
Гражданство	nchar(10)	<input checked="" type="checkbox"/>
Пол	nchar(10)	<input checked="" type="checkbox"/>
Сумма	money	<input checked="" type="checkbox"/>
[Коэффициент выплат]	nchar(10)	<input checked="" type="checkbox"/>
Адрес	nchar(50)	<input checked="" type="checkbox"/>
[Выплаты до]	date	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Таблица (проект)

Ф.И.О.	Дата рождения	Место рожде...	Гражданство	Пол	Сумма	Кoeffициент...	Адрес	Выплаты до
Игнатов Роман Иванович	2012-09-06	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 6-14-123	2030-09-06
Романов Константин Иванов	2011-09-12	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 7-23-22	2029-09-12
Икров Игорь Анатольевич	2012-11-04	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 8-12-23	2030-11-04
Киргорова Анна Андреевна	2010-12-08	г.Егорьевск	РФ	Ж	12000,0000	7,2	г.Егорьевск 5-23-123	2028-04-02
Трапкина Ирина Ивановна	2008-03-06	г.Егорьевск	РФ	Ж	12000,0000	7,2	г.Егорьевск 2-32-231	2008-03-06
Конькова Елизавета Андреевна	2012-09-09	г.Егорьевск	РФ	Ж	12000,0000	7,2	г.Егорьевск 3-10-212	2030-09-09
Колосова Ирина Геннадьевна	2012-11-12	г.Егорьевск	РФ	Ж	12000,0000	7,2	г.Егорьевск 2-12-212	2030-11-12
Лопатин Игорь Андреевич	2012-01-12	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 1-12-123	2030-01-12
Вареников Валентин Игнатович	2012-09-11	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 2-12-213	2030-09-11
Раков Константин Анатольевич	2011-07-04	г.Егорьевск	РФ	М	12000,0000	7,2	г.Егорьевск 1-11-217	2029-04-01

Таблица (список)

Имя столбца	Тип данных	Разрешить ...
[Ф.И.О.]	nchar(50)	<input checked="" type="checkbox"/>
[Дата рождения]	date	<input checked="" type="checkbox"/>
[Место рождения]	nchar(50)	<input checked="" type="checkbox"/>
Гражданство	nchar(10)	<input checked="" type="checkbox"/>
Пол	nchar(10)	<input checked="" type="checkbox"/>
Сумма	money	<input checked="" type="checkbox"/>
[Кoeffициент выплат]	nchar(50)	<input checked="" type="checkbox"/>
Адрес	nchar(50)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Таблица (проект)

Ф.И.О.	Дата рождения	Место рожде...	Гражданство	Пол	Сумма	Кoeffициент...	Адрес
Куприн Олег Иванович	1985-07-03	г.Егорьевск	РФ	М	15000,0000	9	г.Егорьевск 98-54-453
Корниплодов Иван Иванович	1988-11-09	г.Егорьевск	РФ	М	15000,0000	9	г.Егорьевск 45-5-234
Макаров Евгений Антонович	1994-12-05	г.Егорьевск	РФ	М	15000,0000	9	г.Егорьевск 67-34-233
Килькина Анастасия Макарова	1978-09-05	г.Егорьевск	РФ	Ж	15000,0000	9	г.Егорьевск 87-45-12
Крупникова Марина Ивановна	1989-08-08	г.Егорьевск	РФ	Ж	15000,0000	9	г.Егорьевск 34-23-21
Бирюкова Ильвира Акимовна	1988-11-11	г.Егорьевск	РФ	Ж	15000,0000	9	г.Егорьевск 23-25-65
Кандратьев Максим Максимович	1968-02-05	г.Егорьевск	РФ	М	15000,0000	9	г.Егорьевск 65-76-23
Кобалев Макар Викторович	1964-09-10	г.Егорьевск	РФ	М	15000,0000	9	г.Егорьевск 15-65-34
Криков Владимир Олегович	2005-09-09	г.Егорьевск	РФ	М	15000,0000	9	г.Егорьевск 10.14.1
Рыльский Игорь Анатольевич	1945-12-10	г.Егорьевск	РФ	М	15000,0000	9	г.Егорьевск 7-5-6

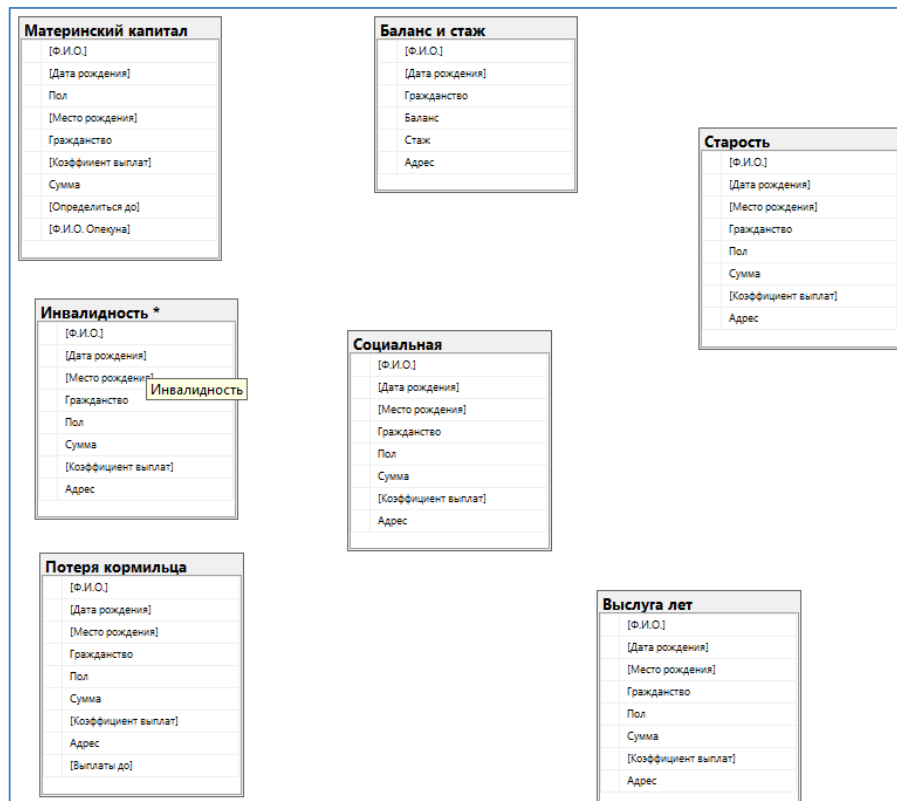
Таблица (список)

Имя столбца	Тип данных	Разрешить ...
[Ф.И.О.]	nchar(50)	<input checked="" type="checkbox"/>
[Дата рождения]	date	<input checked="" type="checkbox"/>
[Место рождения]	nchar(50)	<input checked="" type="checkbox"/>
Гражданство	nchar(10)	<input checked="" type="checkbox"/>
Пол	nchar(10)	<input checked="" type="checkbox"/>
Сумма	money	<input checked="" type="checkbox"/>
[Кoeffициент выплат]	nchar(50)	<input checked="" type="checkbox"/>
Адрес	nchar(60)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

## Таблица (проект)

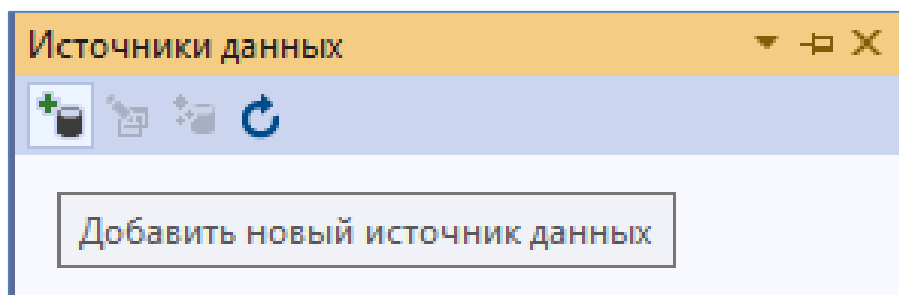
Ф.И.О.	Дата рождения	Место рожде...	Гражданство	Пол	Сумма	Коэффициент...	Адрес
Еромолкина Агния Андреевна	1954-04-08	г.Егорьевск ...	РФ	Ж	12000,0000	7,4	г.Егорьевск 6-14-123
Антонов Игорь Геннадьевич	1953-05-09	г.Егорьевск ...	РФ	М	12000,0000	7,4	г.Егорьевск 6-14-125
Иванов Михаил Артемович	1955-09-12	г.Егорьевск ...	РФ	М	12000,0000	7,4	г.Егорьевск 8-24-23
Романова Елизавета Юрьевна	1956-05-11	г.Егорьевск ...	РФ	Ж	12000,0000	7,4	г.Егорьевск 1-1-1
Коротков Александр Александрович	1950-08-08	г.Егорьевск ...	РФ	М	12000,0000	7,4	г.Егорьевск 7-18-16
Икров Рустам Александрович	1956-03-09	г.Егорьевск ...	РФ	М	12000,0000	7,4	г.Егорьевск 3-11-12
Костоломова Анна Евгеньевна	1956-05-11	г.Егорьевск ...	РФ	Ж	12000,0000	7,4	г.Егорьевск 6-10-11
Круглова Ильвира Степановна	1950-08-08	г.Егорьевск ...	РФ	Ж	12000,0000	7,4	г.Егорьевск 6-10-11
Крысюк Антонина Иванна	1953-05-09	г.Егорьевск ...	РФ	Ж	12000,0000	7,4	г.Егорьевск 6-12-11
Ивлева Ирина Николаевна	1956-05-11	г.Егорьевск ...	РФ	Ж	12000,0000	7,4	г.Егорьевск 7-12-11

## Диаграмма



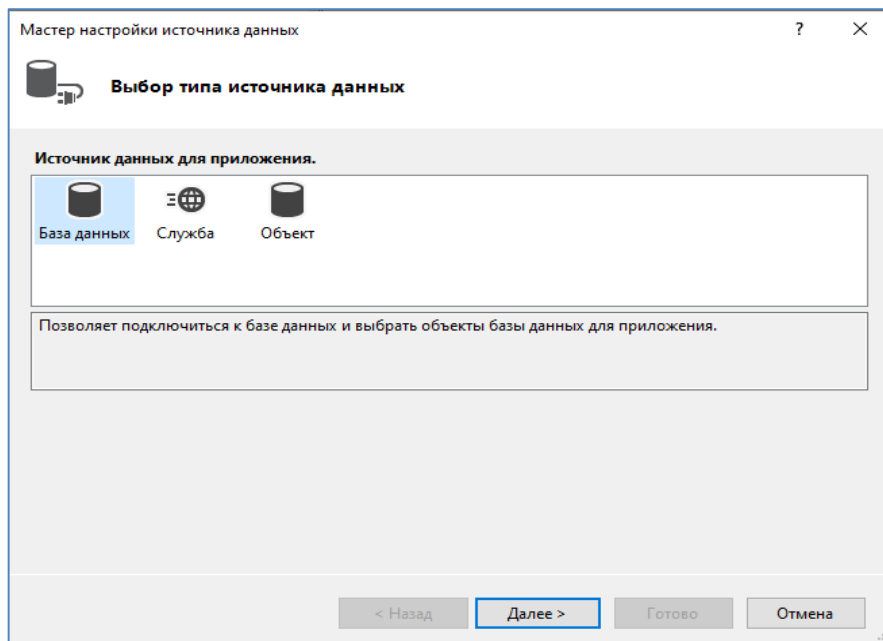
### Привязка БД к Visual Studio:

В начале надо перейти на вкладку «Источник данных» в проекте Visual Studio, нажать кнопку «Добавить новый источник данных» в левом верхнем углу:

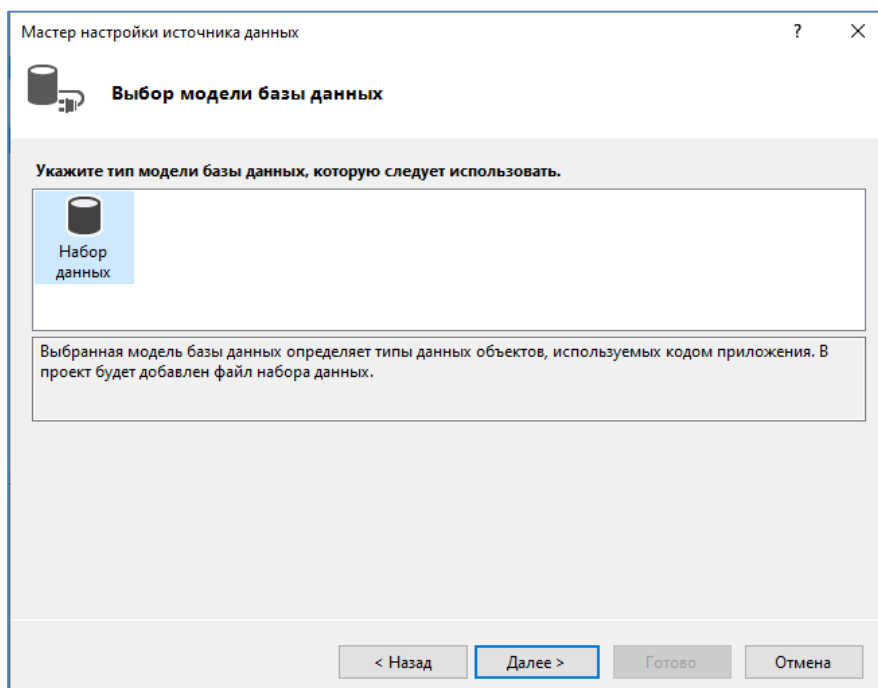


## Добавить новый источник данных

После откроется окно с названием «Мастер настройки источника данных» данное окно имеет следующий вид:



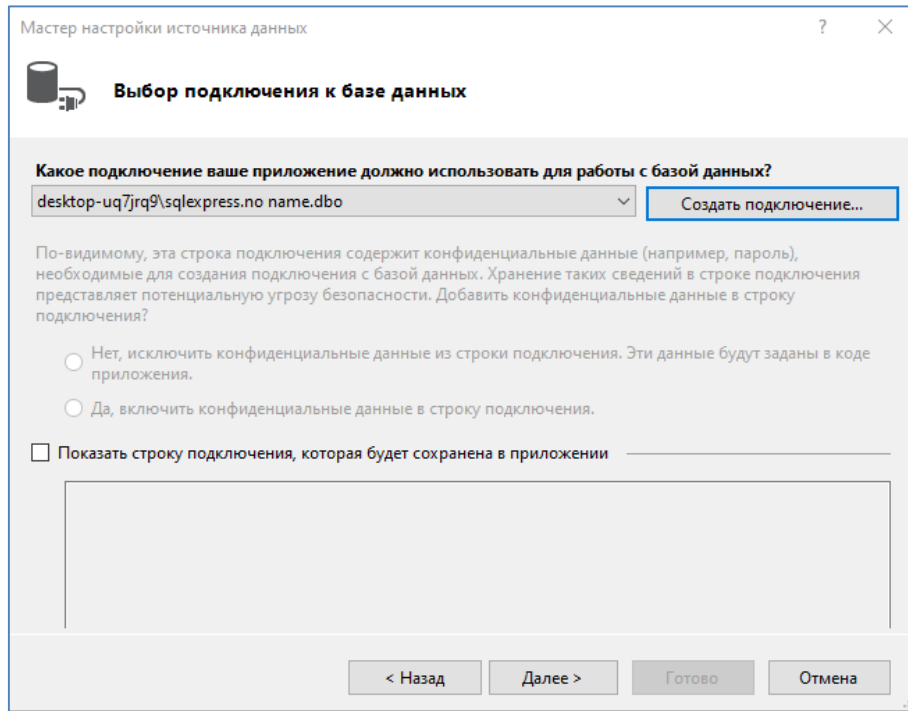
В данном окне нужно выбрать источник данных для приложения в данном случае это база данных кликнув на иконку с названием «База данных» нужно нажать кнопку «Далее» и в результате перекинет на другое окно где надо указать тип модели базы данных мы выбрали тип с названием «Набор данных» и кликнул кнопку «Далее»:



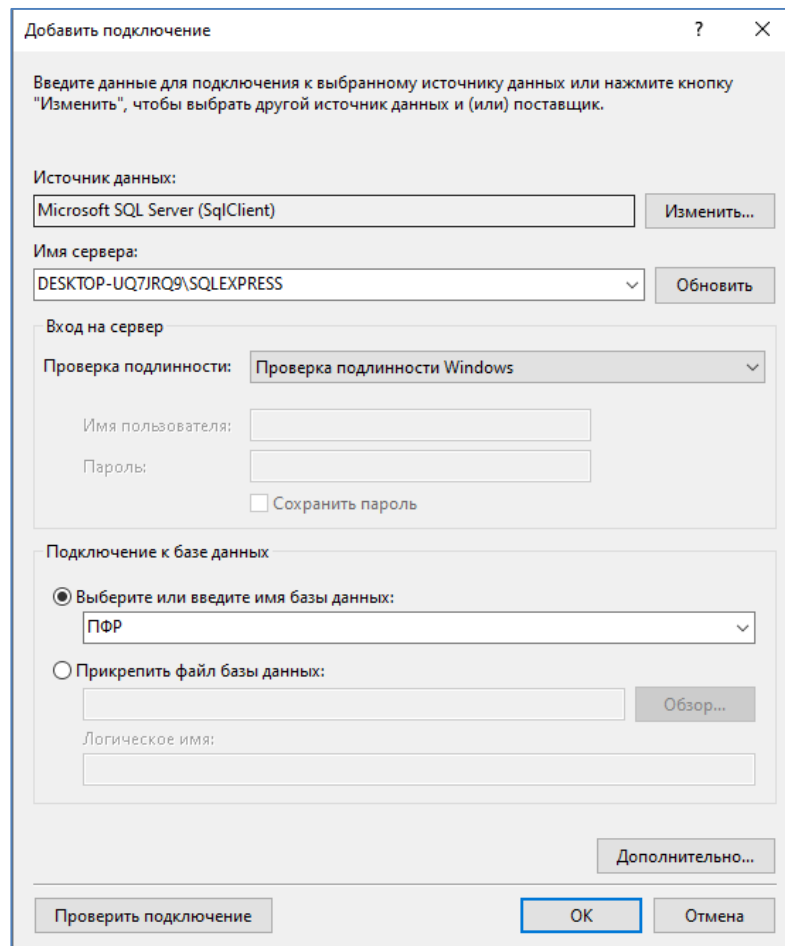
## Мастер настройки источника данных

Потом программа спросит какое подключение ваше приложение должно использовать мы выбрали: «desktop-uq7jr99\sqlexpress.ПФР.dbo» для этого надо нажать кнопку «Создать подключение» после чего перекинет на окно «Добавить подключение» в этом окне надо указать имя сервера и ввести имя базы данных, далее рассмотрим как это выглядит в программе:

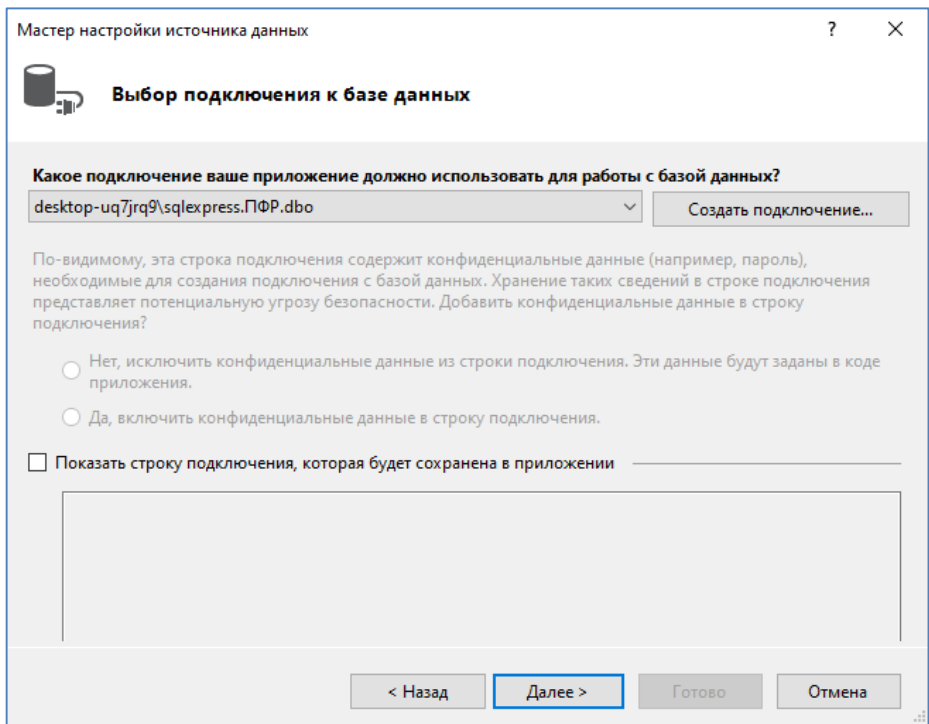
### Выбор подключения к базе данных



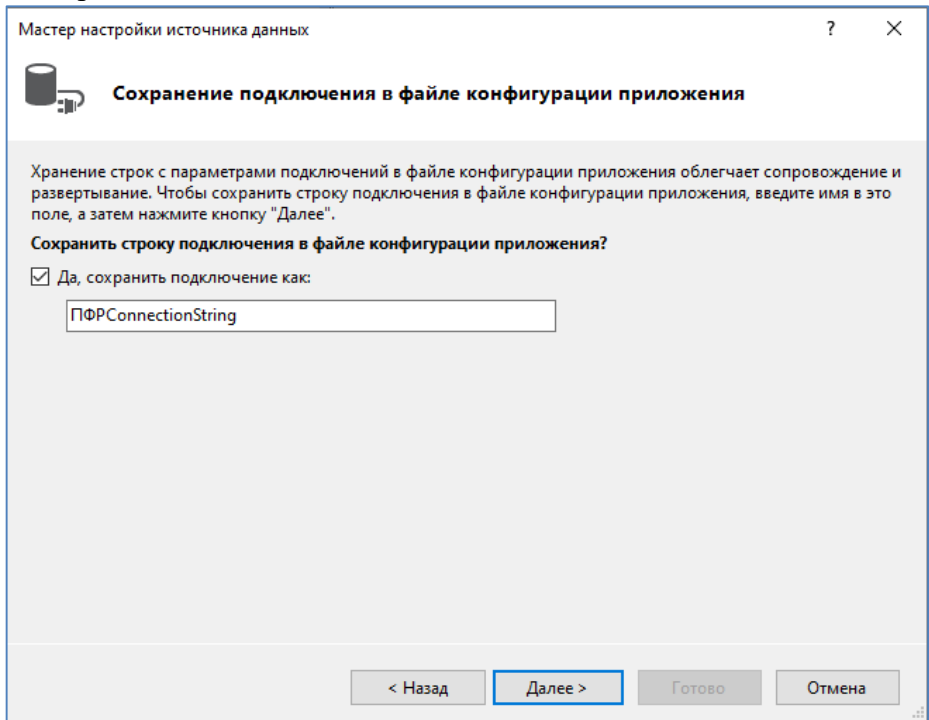
### Добавление подключения



## Выбор подключения к базе данных

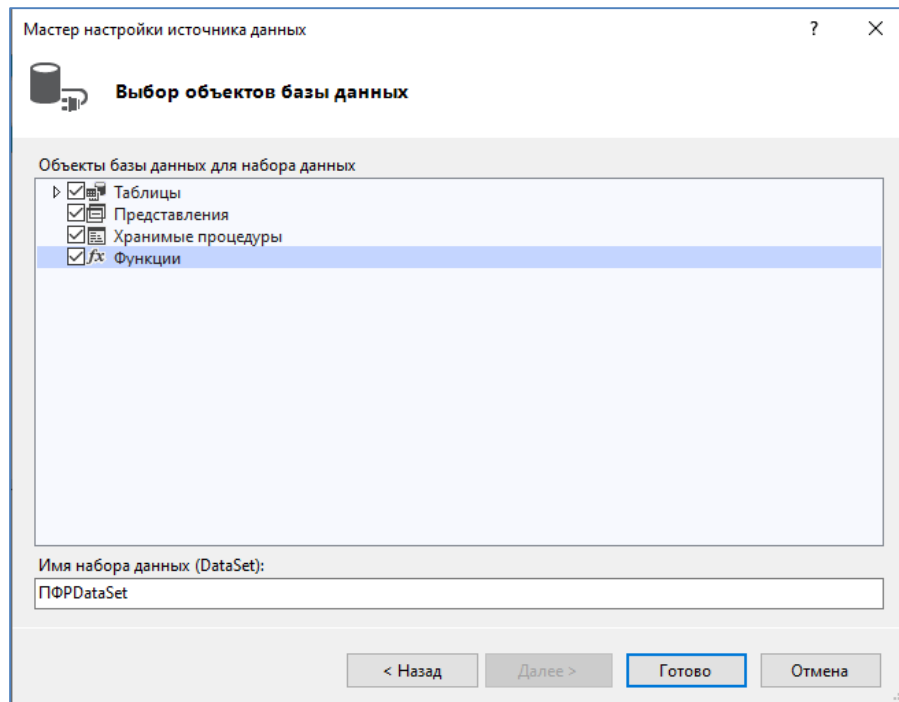


Далее необходимо сохранить подключение:

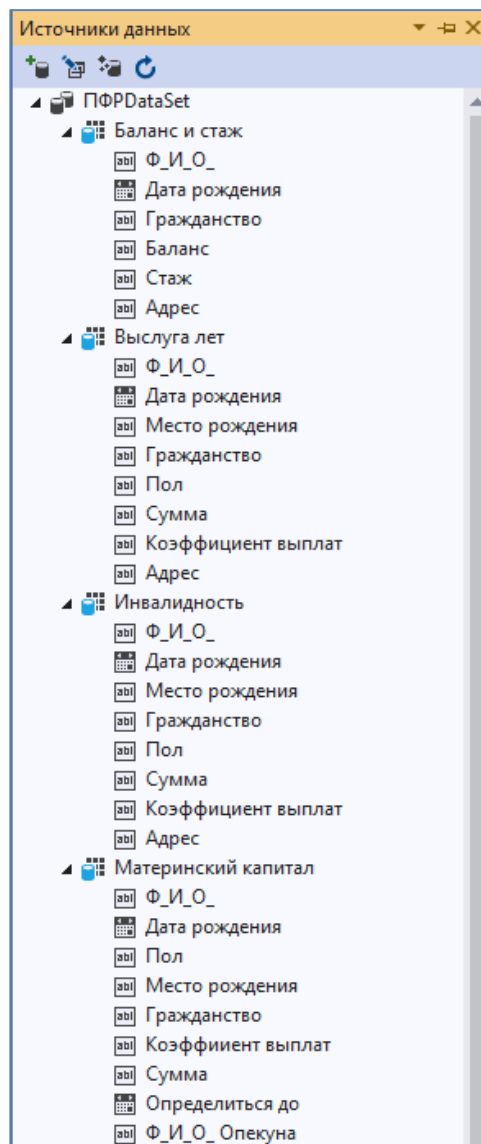


Сохранение подключения

Надо выбрать необходимые объекты БД.



Выбор объектов



Все успешно подключилось:



## Подключение БД

После всех действий выше началось создание проекта в Visual Studio. В Visual Studio добавляются элементы такие как кнопки, панели, метки, поля с рисунками и текстовые поля, а также элементы интерфейса картинки. Еще надо написать код для связи некоторых элементов между собой.

Авторизация:

- Форма содержит все элементы, имеет белый фон;
- Метка содержит текстовую информацию;
- Текстовые поля, одно служит для введения логина а вторая для ввода пароля;
- Кнопка «Вход» имеет код благодаря которому если информация совпадет, вход осуществится, если не совпадет, выдаст окно с предупреждением «Не правильный логин или пароль»;
- Кнопка «Выход» закрывает программу;
- Поле с рисунком содержит рисунок.

Ниже представлены код «Вход» и «Выход» а также вид «Авторизации» в программе:

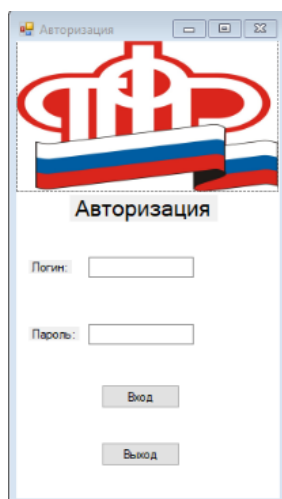
Код «Вход»:

```
private void button1_Click(object sender, EventArgs e)
{
    if (textBox1.Text == "admin")
    {
        if (textBox2.Text == "admin")
        {
            Меню f1 = new Меню();
            f1.ShowDialog();
        }
    }
    else
        MessageBox.Show("Не правильный логин или пароль");
    textBox1.Clear();
    textBox2.Clear();
}
```

Код «Выход»:

```
private void button2_Click_1(object sender, EventArgs e)
{
    this.Close();
}
```

Авторизация



Меню:

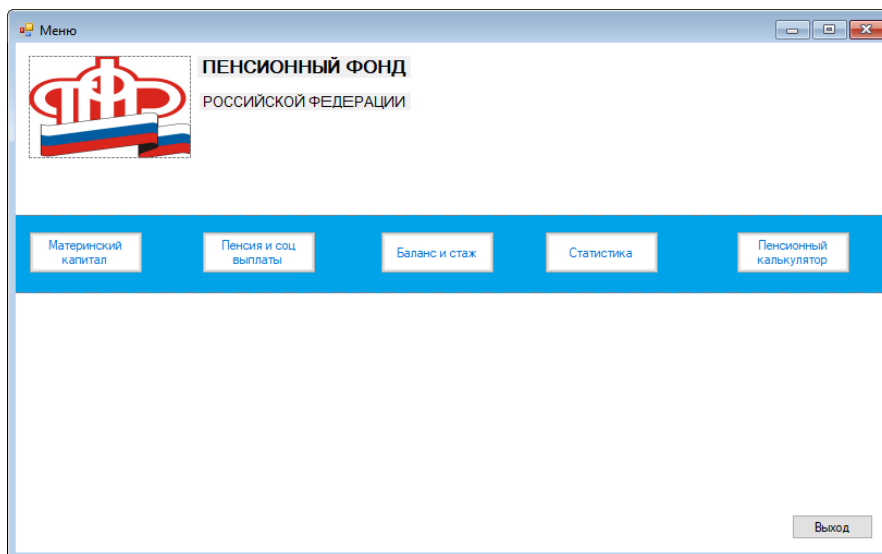
- Форма содержит все элементы, имеет белый фон;
- Метка содержит текстовую информацию;
- Поле с рисунком содержит рисунок;
- Форма содержит панель на которой расположены кнопки(имеет голубой фон)
- Кнопки их всего 6 и 5 из них позволяют перейти на другую форму, 1 дает возможность выхода из программы.

Ниже представлены код и вид формы «Меню»:

Кнопки перехода:

```
private void button1_Click(object sender, EventArgs e)
{
    Hide();
    Form1 = new Form1();
    .ShowDialog();
}
```

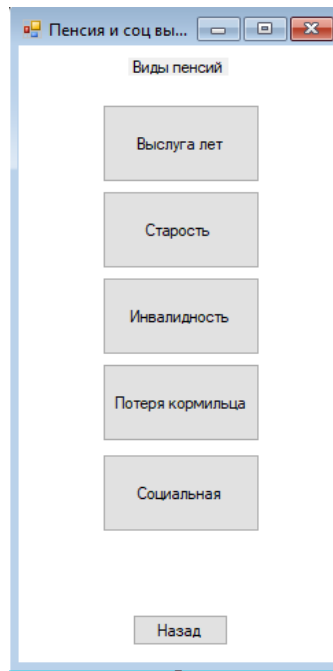
Меню



Пенсия и соц выплаты:

- Форма содержит все элементы, имеет белый фон;
- Метка содержит текстовую информацию;
- 6 кнопок перехода между формами.

Вид формы «Пенсия и соц выплаты»:



Материнский капитал:

- Форма содержит все элементы, имеет белый фон;
- Тестовое поле с данными;
- Кнопка «Сохранить» и кнопка «Назад»

Код кнопки «Сохранить» и вид формы «Материнский капитал»:

Код «Сохранить»:

```
using ПФР.Properties;
```

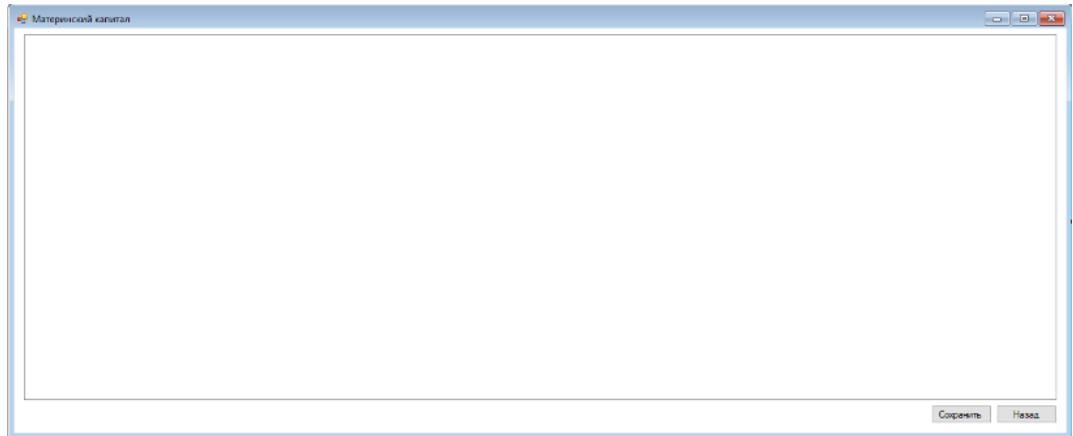
```

.....
InitializeComponent();
textBox1.Text = Settings.Default.textBox1;
.....
private void button2_Click_1(object sender, EventArgs e)
{
    Settings.Default.textBox1 = textBox1.Text;
    Settings.Default.Save();
    textBox1.Text = Settings.Default.textBox1;
}

```

Параметры

	Имя	Тип	Область.	Значение
	textBox1	string	Пользователь	
	textBox2	string	Пользователь	
	textBox3	string	Пользователь	
	textBox4	string	Пользователь	
	textBox5	string	Пользователь	
	textBox6	string	Пользователь	
	textBox7	string	Пользователь	
	textBox8	string	Пользователь	
*				



#### 4. Оптимизация и рефакторинг кода

Были заданы следующие переменные для удобства работы с кодом:

```
Ссылка: 4  
public partial class Form4 : Form  
{  
    public string D;  
    public string N1;  
    public bool n2;  
}
```

Переменные

Были добавлены следующие классы:

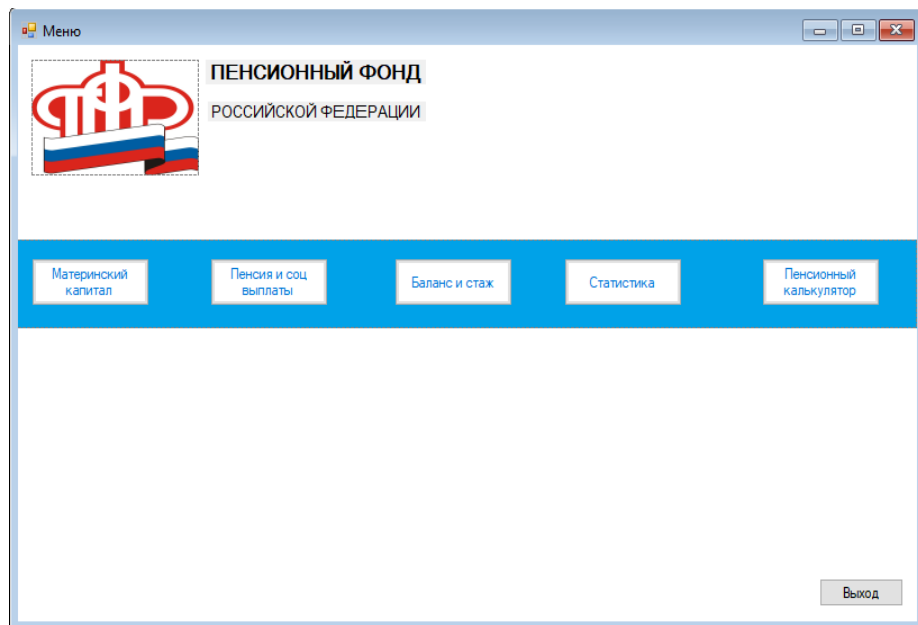
```
using System.Threading.Tasks;  
using System.Windows.Forms;  
using ПФР.Properties;
```

Классы

#### 5. Разработка интерфейса программного модуля

При разработке интерфейса были учтены все указанные требования (шрифт, цвет и прочее).

Вид программы:



## 6. Тестирование программного модуля

Тестирование проводилось одновременно с проектированием и разработкой логики программного модуля.

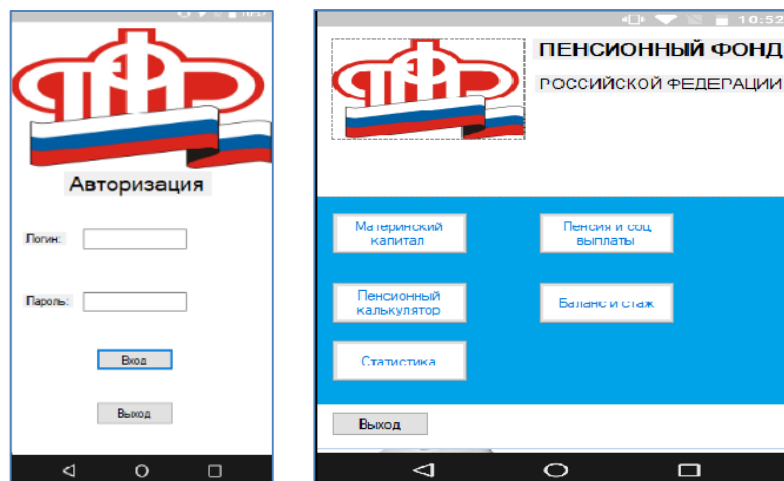
Для этого были по умолчанию заполнены данными необходимые элементы управления. Тестирование авторизации показало что ПО способно распознать того кто использует программу, для этого был использован блок кода.

Все кнопки, функции и списки работают исправно.

## 7. Проектирование мобильного приложения. Написание кода мобильного приложения.

Разработка мобильного приложения в целом аналогична разработке оконного приложения, я лишь покажу что вышло в итоге.

Вид



## 8. Оптимизация и рефакторинг кода мобильного приложения. Разработка интерфейса мобильного приложения. Тестирование мобильного приложения.

Так как при разработке оконного приложения уже было проведено тестирование, рефакторинг, оптимизация и был оформлен интерфейс то я лишь скопировал его в мобильное приложение.

## 9. Оформление документации пользователя

### О программе

Программа предназначена для взаимодействия с таблицей (редактирование, добавление, удаление записей), просмотра информации пенсии и других соц выплат а также был добавлен калькулятор.

### Форма авторизации

На форме размещены поля для ввода логина и пароля, кнопки входа и выхода.

### Форма меню

Ключевая форма, на форме размещены кнопки для перехода между всеми окнами и кнопка выхода.

### Форма материнский капитал

На форме размещены поле с данными кнопка «Сохранить» и кнопка «Назад».

### Форма пенсионный калькулятор

На форме размещены кнопки поле для ввода информации и вывода новой при контакте с определенными кнопками.

*Остальные формы похожи на форму «Материнский капитал».*

Производственная практика является одним из основных условий закрепления полученных в техникуме теоретических знаний, приобретения практических навыков по их применению, а также выявления пробелов в знаниях теории. Практика дает возможность получить практическую подготовку, оценить свои возможности. Считаю, что программу практики выполнил в полном объеме, получив необходимые практические навыки и закрепив полученные теоретические знания.

Создав данный проект «Информационная система ПФР» я укрепил теоретические знания полученные в техникуме а также приобрел новые знания к примеру благодаря решению ошибок в программе которые появлялись по ходу создания проекта, стал лучше понимать SQL и Visual Studio.

Доклад на тему:

## **«Проектирование программного модуля для автоматизации работы библиотечной системы»**

### Место прохождения практики:

Центральная библиотека г. Егорьевска

Организационная структура предприятия была устроена следующим образом:

Для выполнения работы я использовал свой персональный компьютер:

Видео карта: GeForce RTX 2080,

Процессор: i7-8700 CPU 3.20Ghz 3.19 GHz,

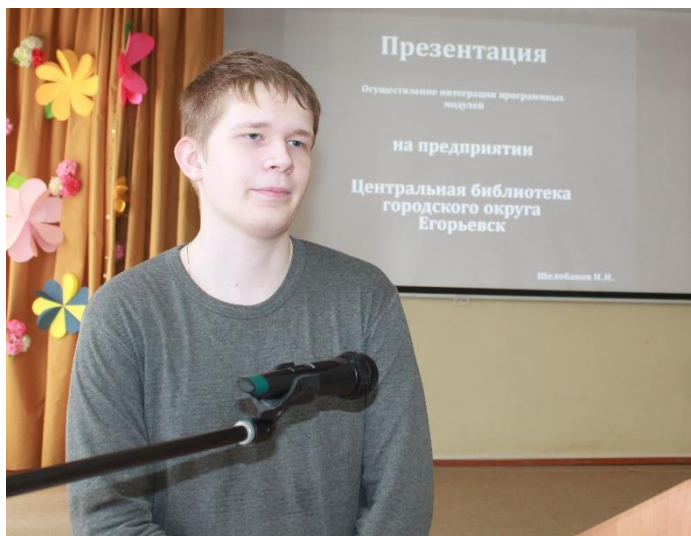
Оперативная память: 16 Гб.

Программное обеспечение на ПК:

Windows 10;

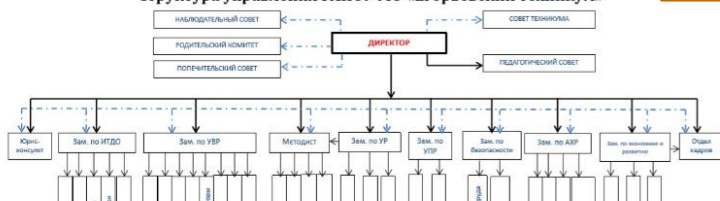
Стандартный пакет офисов Microsoft;

Visual Studio;



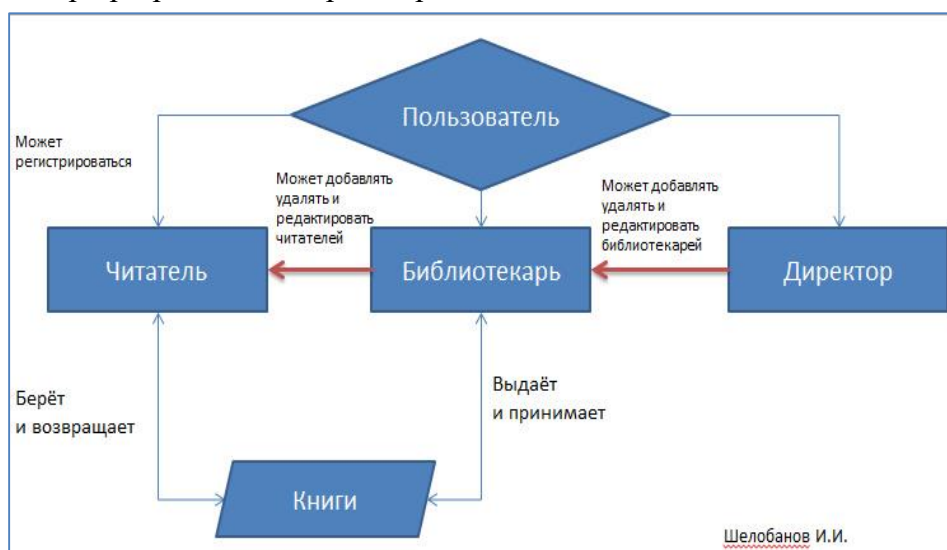
**Шелобанов Иван – студент группы Ип-73**

### Структура управления ГАПОУ МО «Егорьевский техникум»



## Разработка алгоритма решения поставленной задачи. Анализ требований программного обеспечения. Определение спецификации программного обеспечения

При разработке алгоритма решения была составлена схема:



Из неё следует, что в программе есть «Пользователь», который может иметь одну из трёх ролей: «Читатель», «Библиотекарь», «Директор». Основной функционал будет реализован на взаимодействии пользователей «Читатель» и «Библиотекарь» с книгами. А также, на взаимодействии пользователей друг с другом.

Читатель может регистрироваться, но не может удалить свою учётную запись самостоятельно, а также не может удалить или зарегистрировать «Библиотекаря» и «Директора». Читатель может брать и возвращать книги.

Библиотекарь может удалять, добавлять и редактировать данные читателей. Библиотекарь не может зарегистрировать нового «Библиотекаря», или «Директора». Библиотекарь может выдавать и принимать книги.

Директор может добавлять, удалять и редактировать «Библиотекаря».

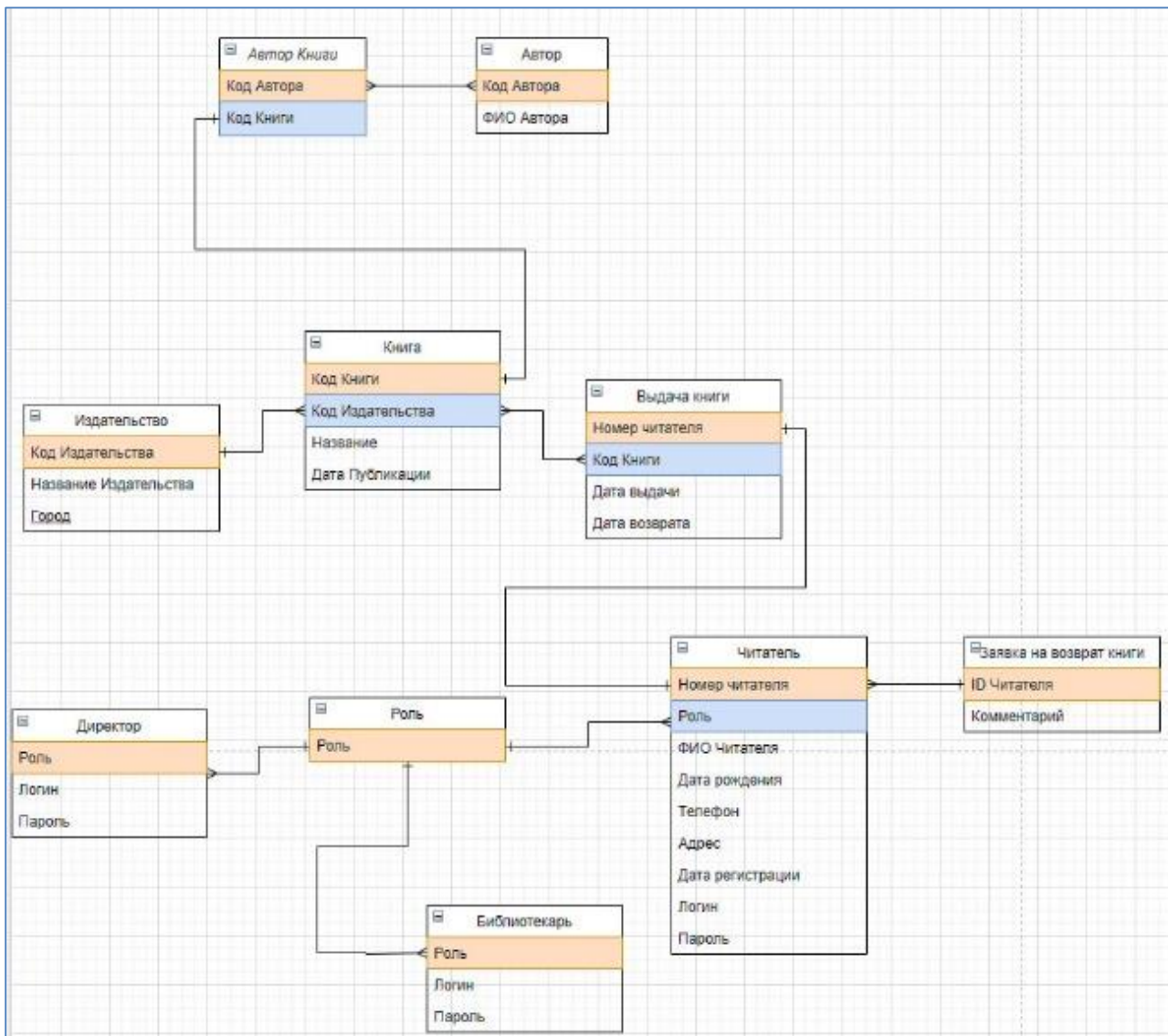
При анализе требований к программному обеспечению было установлено, что для корректной работы программы, требуется:

- Операционная система Windows;
- Сервер SQL;
- 150 МБ памяти на диске;

При определении спецификации программного обеспечения было определено, что программа будет информационной системой.

### Проектирование программного модуля в соответствии с заданием. Добавление выполненной работы в систему контроля версий

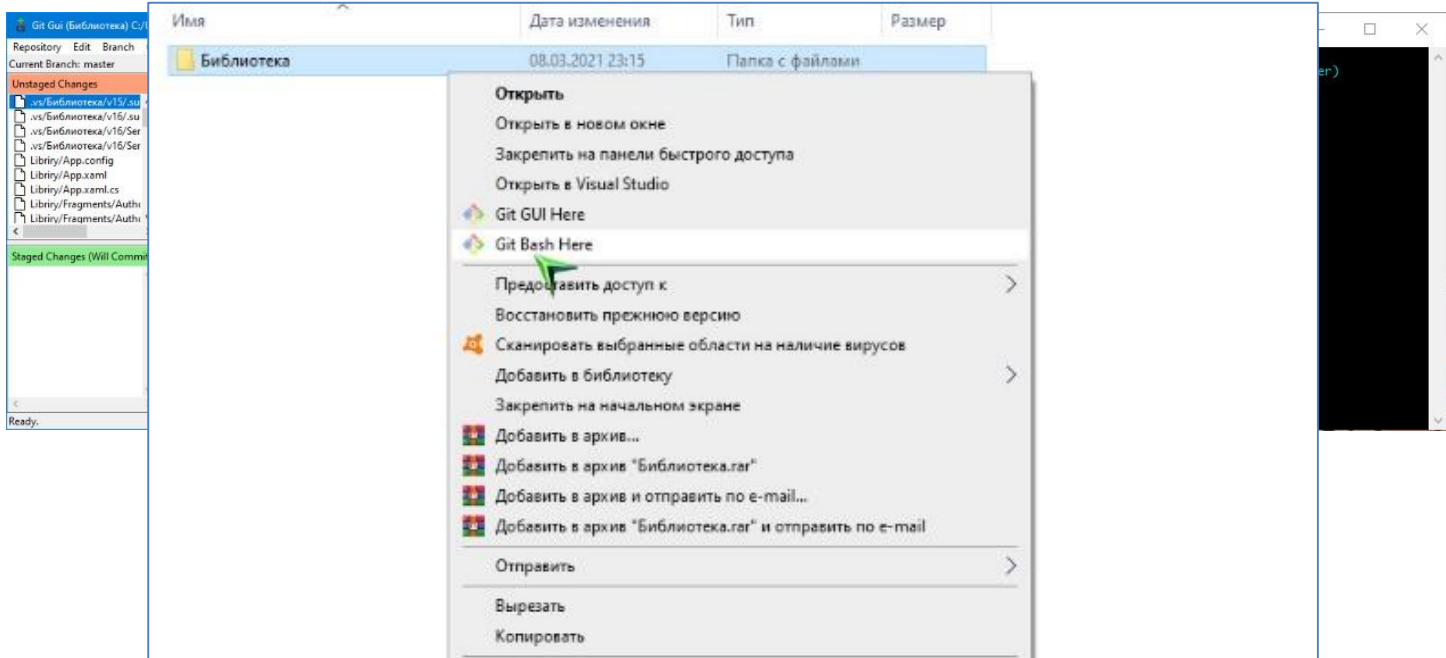
Далее в соответствии с заданием была спроектирована схема программного модуля:





После завершения работы с программой, она была добавлена в систему контроля версий

Git:



## Программирование программного обеспечения в соответствии с заданием.

В соответствии с заданием был написан код для программного модуля, а также созданы страницы и классы.



```
var Login = LoginBox.Text;
var Password = PasswordBox.Password;
var ResultReader = IDModelLibrary.Reader.FirstOrDefault(a => a.Login == Login && a.Password == Password);
var ResultLibrary = IDModelLibrary.Library.FirstOrDefault(a => a.Login == Login && a.Password == Password);
var ResultDirector = IDModelLibrary.Director.FirstOrDefault(a => a.Login == Login && a.Password == Password);
try
{
    if (ResultReader == null)
    {
        if (ResultLibrary == null)
        {
            if (ResultDirector == null)
            {
            }
            else
            {
                if (ResultDirector.Role == "Директор")
                {
                    MessageBox.Show("Вы директор!");
                }
            }
            else
            {
                if (ResultLibrary.Role == "Библиотекарь") { NavigationService.Navigate(new AuthorLibrary()); }
            }
        }
        else
        {
            if (ResultReader.Role == "читатель") { NavigationService.Navigate(new PageReader()); }
        }
    }
}
catch
{
    MessageBox.Show("Неверный логин или пароль!");
}
```

```
Books_Code.ItemsSource = Fragments.IDModelLibrary.Books.ToList();

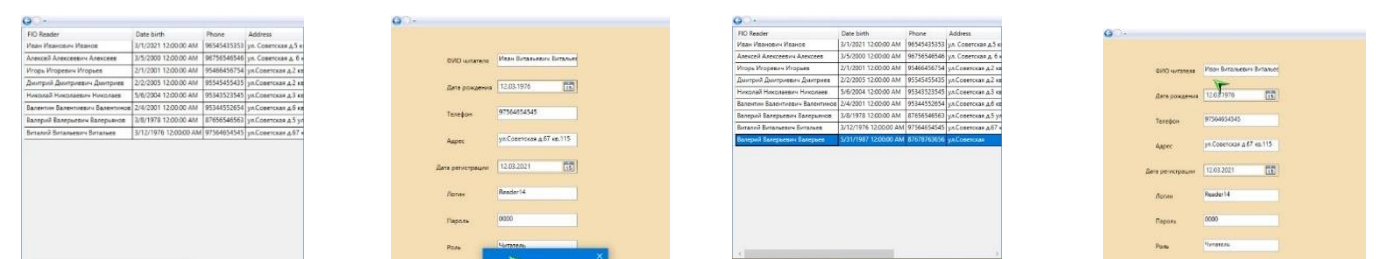
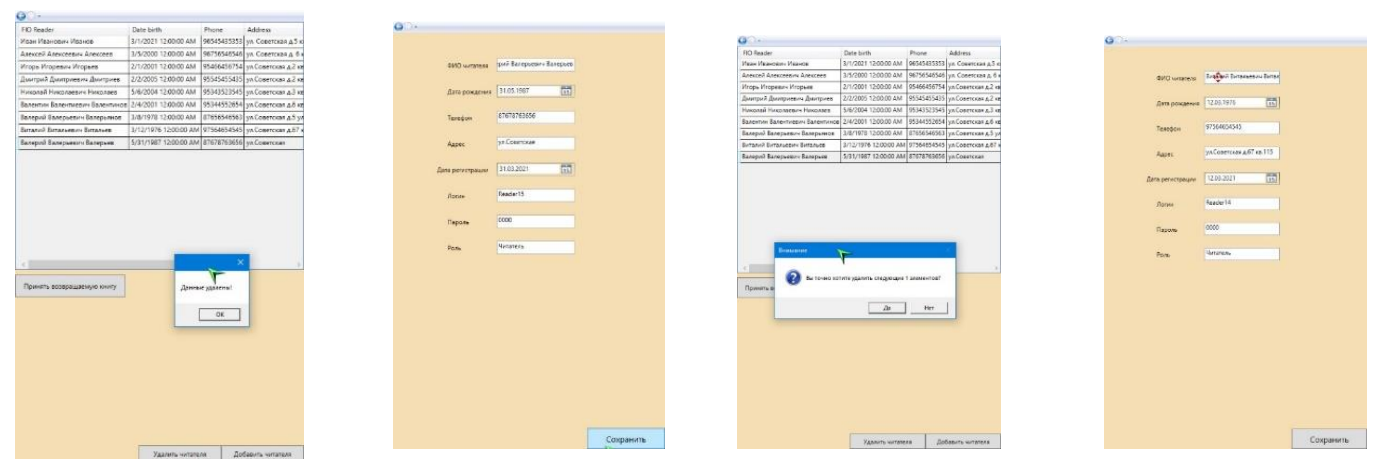
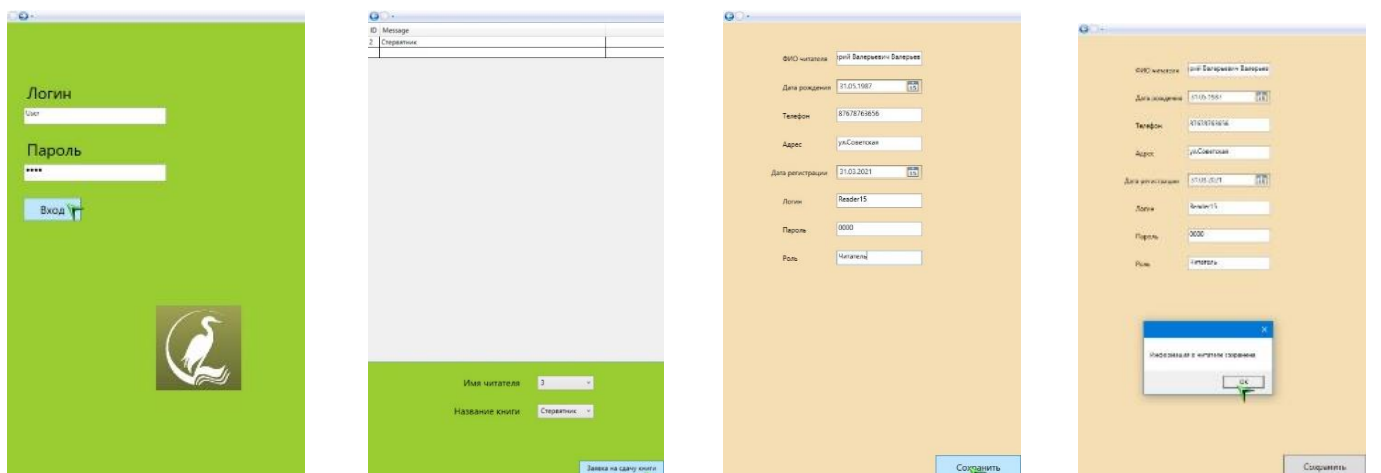
private void DatePicker_DataContextChanged(object sender, DependencyPropertyChangedEventArgs e)
{
}

private void DatePicker_Loaded(object sender, RoutedEventArgs e)
{
    DatePicker.SelectedDate = DateTime.Now;
}

private void Button_Click(object sender, RoutedEventArgs e)
{
    try
    {
        Extraction EQ = new Extraction();
        EQ.Reader_ticket = Convert.ToInt32(Code_Book.Text);
        EQ.Code_Book = Convert.ToInt32(Book_Code.Text);
        EQ.Date_of_capture = DatePicker.SelectedDate;
        LabelLine.Content = "Время выкуры " + DatePicker.SelectedDate.Value.AddMonths(1).ToString() + " мес!";
        EQ.Delivery_period = DatePicker.SelectedDate.Value.AddMonths(1);
        IDModelLibrary.Extraction.Add(EQ);
    }
}
```

## Тестирование программного обеспечения.

При программировании были созданы страницы перехода, окно авторизации и регистрации, а также реализован остальной функционал приложения:



code	Address	header status	Register date	
14541333	ул. Советская д.3 кв.3	1	3/1/2021 12:00:00 AM	Радужная
79549348	ул. Советская д. 6 кв.2	2	3/1/2021 12:00:00 AM	Радужная
48620728	ул. Советская д.1 кв.2	2	3/1/2021 12:00:00 AM	Радужная
54245245	ул. Советская д.1 кв.2	4	3/1/2021 12:00:00 AM	Радужная
48132345	ул. Советская д.1 кв.2	3	3/1/2021 12:00:00 AM	Радужная
34470194	ул. Советская д.8 кв.1	8	3/1/2021 12:00:00 AM	Радужная
10043033	ул. Советская д.1 кв.2	1018	3/1/2021 12:00:00 AM	Радужная
56464545	ул. Советская д.7 кв.112	1021	3/1/2021 12:00:00 AM	Радужная

BookCode	BookCode	BookCode	BookCode
1	2	3/12/2021 4:59:51 PM	4/12/2021 4:59:51 PM
1	2	3/12/2021 4:59:51 PM	4/12/2021 4:59:51 PM

FKO Number	Date birth	Phone	Address
Ром Радужный Романа	3/1/2021 12:00:00 AM	9174541333	ул. Советская д.1
Дмитрий Дмитриевич Дмитриев	3/1/2021 12:00:00 AM	9174541333	ул. Советская д.1
Роман Радужный Романа	3/1/2021 12:00:00 AM	9174541333	ул. Советская д.1
Дмитрий Дмитриевич Дмитриев	3/1/2021 12:00:00 AM	9174541333	ул. Советская д.1
Владимир Владимирович Владимир	3/1/2021 12:00:00 AM	9174541333	ул. Советская д.1
Владимир Владимирович Владимир	3/1/2021 12:00:00 AM	9174541333	ул. Советская д.1
Владимир Владимирович Владимир	3/1/2021 12:00:00 AM	9174541333	ул. Советская д.1
Владимир Владимирович Владимир	3/1/2021 12:00:00 AM	9174541333	ул. Советская д.1

BookCode	BookCode	BookCode	BookCode
1	2	3/12/2021 4:59:51 PM	4/12/2021 4:59:51 PM
1	2	3/12/2021 4:59:51 PM	4/12/2021 4:59:51 PM

### Оптимизация и рефакторинг кода

Для оптимизации работы приложения мой был написан паттерн «Одиночка», благодаря которому можно за место того, чтобы прописывать код работы с данными в каждой страницы, просто вызвать нужный метод из паттерна.

```

public partial class LibraryEntities1 : DbContext
{
    private static LibraryEntities1 _context;
    Ссылка: 2
    public LibraryEntities1()
        : base("name=LibraryEntities1")
    {
    }

    Ссылка: 0
    protected override void OnModelCreating(DbModelBuilder modelBuilder)
    {
        throw new UnintentionalCodeFirstException();
    }

    Ссылка: 4
    public static LibraryEntities1 GetContext()
    {
        if (_context == null)
            _context = new LibraryEntities1();
        return _context;
    }
}

```

### Оформление документации пользователя

При использовании программы, читателям необходимо быть ознакомленными с основами регистрации и авторизации. Ни в коем случае не терять записанные логин и пароль, а при возникновении утраты подобной информации немедленно связаться с библиотекарем.

Библиотекарям же следует компетентно обращаться с информацией, что занесена в информационную систему, а также быть внимательными при регистрации новых читателей. В случае, если данные о читателях или книгах были введены неправильно, требуется немедленно исправить неточности в данных. Библиотекарям так же не рекомендуется предоставлять доступ, лицу, не относящемуся к библиотечному составу или руководству, к другим читателям.

Программа позволяет библиотекарю: регистрировать новых читателей и принимать взятые книги. Для каждого вида пользователя существует своё отдельное пространство, со своим списком предоставляемых возможностей.

Библиотекарь может работать с информацией о книгах и читателях. Читатели могут выбирать и брать книги. А директор может добавлять или удалять библиотекарей.

В случае каких-либо сбоев в программе, или же отказов в работе рекомендуется перезагрузить устройство, на котором установлено приложение. В случае, если данный способ

не помог, следует переустановить приложение. Отчёты по работе приложения, при возможности, необходимо отправлять разработчику, для улучшения качества программных продуктов.

С данными занесёнными в программу, следует обращаться с осторожностью. Разработчик не несёт ответственности, если по вине какого-либо лица, подпущенного к работе, с вносимой информацией, будут удалены все внесённые ранее данные или внесены новые аккаунты, которых в базе быть не должно.

Доклад на тему:

## «Разработка программного модуля «Видеонаблюдение»

**Место прохождения практики:**

Компания «U-VID.RU», г. Егорьевск

Проект предназначен для более удобного заказа услуг по определённому адресу и возможности оставлять отзывы о проделанной работе по установке, обновлению и демонтажу видеонаблюдения. Также даёт возможность администратору просматривать и редактировать заказы и отзывы.

Цель: Разработать программный модуль "Видеонаблюдение"

Задачи:

1. Разработать форму авторизации
2. Разработать форму регистрации нового администратора
3. Разработать форму редактирования клиентов
4. Разработать форму редактирования отзывов
5. Разработать форму выбора услуг и оформления заказа
6. Разработать форму отзывов

Исходя из изложенных выше цели и задач, были выбраны следующие средства разработки: ИСР (интегрированная среда разработки) «VisualStudio 2019 Community», в качестве языка программирования выбран C#, с использованием технологии WPF.netFramework и ADO.NET.

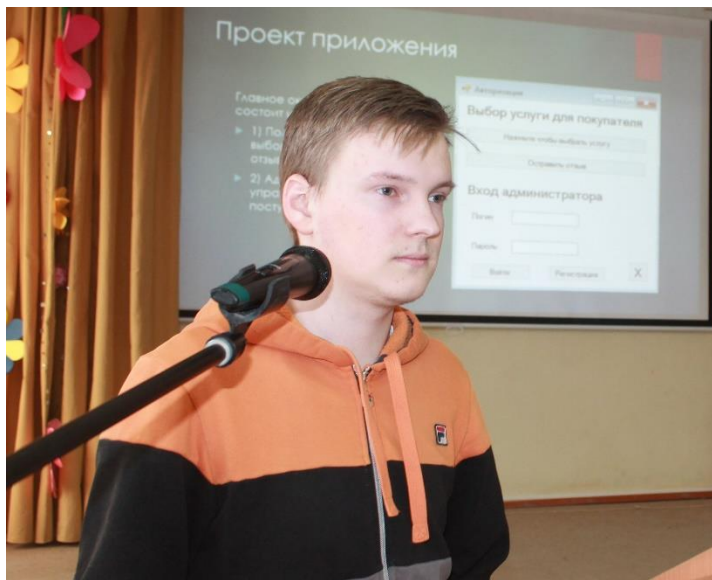
Выбор ИСР обусловлен личными предпочтениями к VisualStudio, а конкретнее привычкой и наибольшими навыками использования данной среды в сравнении с другими.

Выбор языка программирования C# обусловлен наибольшими навыками и опытом программирования на данном языке в сравнении с другими языками.

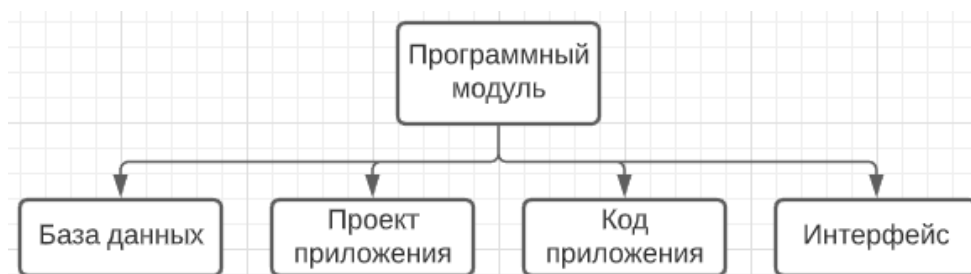
Выбор технологии WPF.netFramework обусловлен, тем что WPF имеет достаточный для разработки модуля набор инструментов и большими возможностями в создании эстетически приятного интерфейса.

### РАЗРАБОТКА АЛГОРИТМА РЕШЕНИЯ ПОСТАВЛЕННОЙ ЗАДАЧИ

При разработке программного модуля «Видеонаблюдение» была составлена диаграмма, изображающая структуру всего проекта, показанная на *Рисунке 1* представленном ниже.



**Гусев Артём –**  
студент группы Ип-73



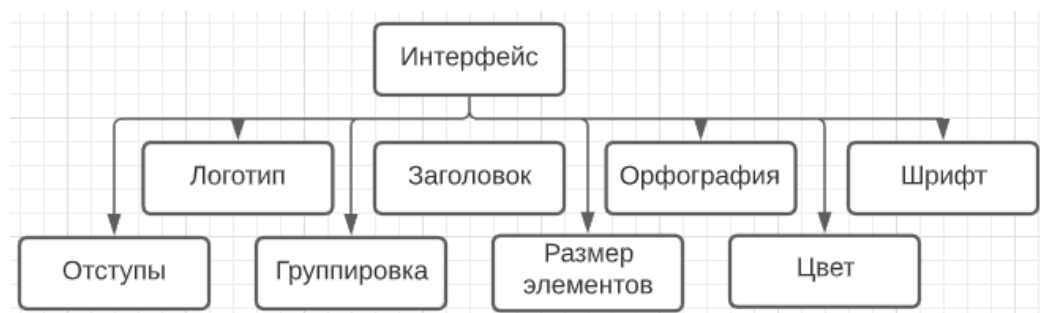
*Рисунок 1. Программный модуль*

При разработке программного модуля «Видеонаблюдение» была составлена диаграмма, изображающая структуру создания базы данных, показанная на *Рисунке 2* представленном ниже.



*Рисунок 2. База данных*

При разработке программного модуля «Видеонаблюдение» была составлена диаграмма, изображающая структуру интерфейса приложения, показанную на *Рисунке 3* представленном ниже.



*Рисунок 3. Интерфейс приложения*

При разработке программного модуля «Видеонаблюдение» была составлена диаграмма, изображающая структуру проекта приложения, показанную на *Рисунке 4* представленном ниже.

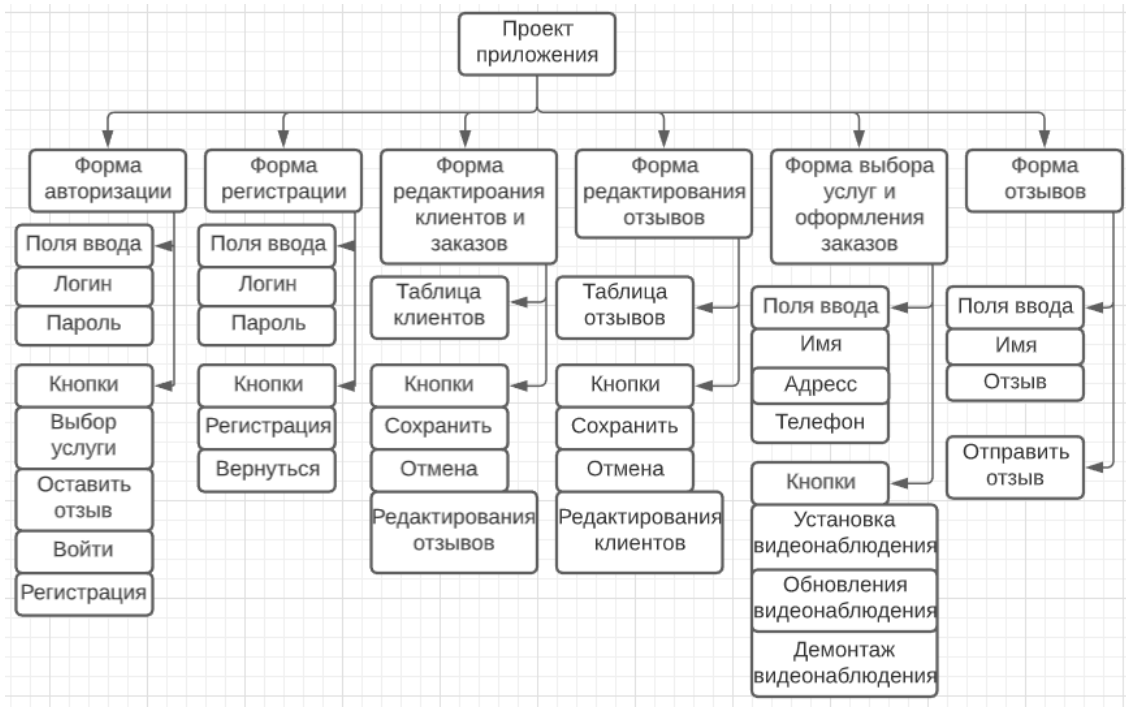


Рисунок 4. Проект приложения

При разработке программного модуля «Видеонаблюдение» была составлена диаграмма, изображающая структуру кода приложения, показанную на *Рисунке 5* представленном ниже.

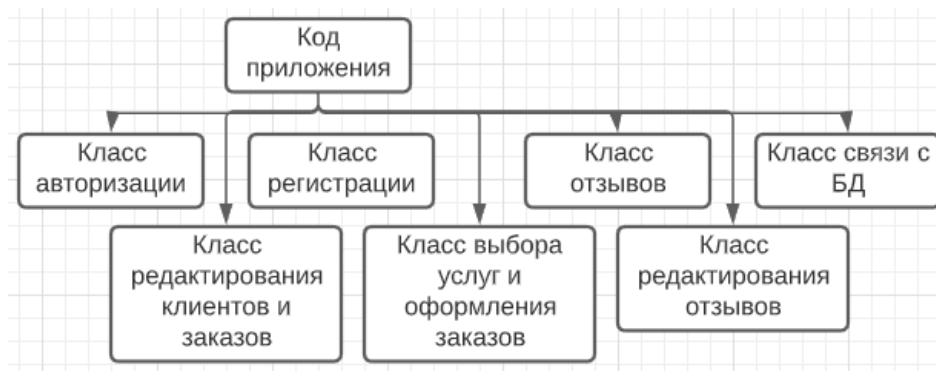


Рисунок 5. Код приложения

## ПРОЕКТИРОВАНИЕ ПРОГРАММНОГО МОДУЛЯ

Форма «Авторизация»

Форма «Регистрация нового Администратора»

### Форма «Редактирования клиентов и заказов»

The screenshot shows a window titled 'Клиенты' with a sub-header 'Редактирование клиентов'. It features a table with columns: id, Услуга, Имя, Телефон, and Адрес. The table is currently empty, with a single row containing an asterisk. Below the table are buttons for 'Сохранить', 'Отмена', and 'Вернуться'.

id	Услуга	Имя	Телефон	Адрес
*				

### Форма «Редактирования отзывов»

The screenshot shows a window titled 'Отзывы' with a sub-header 'Редактирование отзывов'. It features a table with columns: id, Имя, and Отзыв. The table is currently empty, with a single row containing an asterisk. Below the table are buttons for 'Сохранить', 'Отмена', and 'Вернуться'.

id	Имя	Отзыв
*		

### Форма «Выбора услуг и оформления заказа»

The screenshot shows a window titled 'Выбор услуги'. It is divided into two sections: 'Выберите услугу' and 'Оформление заказа'. The first section contains three buttons: 'Установка видеонаблюдения', 'Обновление видеонаблюдения', and 'Демонтаж видеонаблюдения'. The second section contains three input fields labeled 'Укажите имя', 'Укажите телефон', and 'Укажите адрес', followed by an 'Оформить' button.

**Выберите услугу**

Установка видеонаблюдения

Обновление видеонаблюдения

Демонтаж видеонаблюдения

**Оформление заказа**

на

Укажите имя

Укажите телефон

Укажите адрес

Оформить

ПРОГРАММИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

Код Формы «Авторизация»

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Vid
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        class DB
        {
            public static VidioEntities connection = new VidioEntities();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            if ((textBox1.Text != "") && (textBox2.Text != ""))
            {
                var email = textBox1.Text;
                var par = textBox2.Text;
                var result1 = DB.connection.admin.Where(admin => admin.login == email &&
admin.pass == par).FirstOrDefault();
                if (result1 != null)
                {
                    za z = new za();
                    z.Show();
                    this.Hide();
                }
                else
                    MessageBox.Show("Неправильный логин или пароль");
            }
            else MessageBox.Show("Заполните все поля");
        }

        private void button2_Click(object sender, EventArgs e)
        {
            if ((textBox1.Text != "") && (textBox2.Text != ""))
            {
                var email = textBox1.Text;
```



```

        var par = textBox2.Text;
        var result1 = DB.connection.admin.Where(admin => admin.login == email &&
admin.pass == par).FirstOrDefault();
        if (result1 != null)
        {
            reg r = new reg();
            r.Show();
            this.Hide();
        }
        else
            MessageBox.Show("Неправильный логин или пароль");
    }
    else MessageBox.Show("Заполните все поля");
}

```

```

private void button3_Click(object sender, EventArgs e)
{
    us u = new us();
    u.Show();
    this.Hide();
}

```

```

private void button4_Click(object sender, EventArgs e)
{
    ot z = new ot();
    z.Show();
    this.Hide();
}

```

```

private void button5_Click(object sender, EventArgs e)
{
    Application.Exit();
}
}

```

Код Формы «Регистрация нового Администратора»

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

```

```

namespace Vid
{
    public partial class reg : Form
    {
        public reg()
        {
            InitializeComponent();
        }
    }
    class DB

```

```

    {
        public static VidioEntities connection = new VidioEntities();
    }
    private void button2_Click(object sender, EventArgs e)
    {
        if ((textBox1.Text != "") && (textBox2.Text != ""))
        {
            SqlConnection con = new SqlConnection(@"Data Source=DESKTOP-
62J7H61\SQLEXPRESS;Initial Catalog=Vidio;Integrated Security=True");
            con.Open();
            SqlCommand cmd = new SqlCommand("insert into admin(login,pass)
values("+textBox1.Text+", "+textBox2.Text+")",con);
            cmd.ExecuteNonQuery();
            MessageBox.Show("Выполнено");
        }
        else MessageBox.Show("Заполните все поля");
    }

    private void button1_Click(object sender, EventArgs e)
    {
        Form1 f = new Form1();
        f.Show();
        this.Close();
    }

    private void button5_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }
}

```

Код Формы «Редактирования клиентов и заказов»

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace Vid
{
    public partial class za : Form
    {
        public za()
        {
            InitializeComponent();
        }

        private void button2_Click(object sender, EventArgs e)
        {
            kl m = new kl();
            m.Show();
            this.Close();
        }
    }
}

```

```
}
```

```
private void button6_Click(object sender, EventArgs e)
{
    Form1 f = new Form1();
    f.Show();
    this.Close();
}
```

```
private void button1_Click(object sender, EventArgs e)
{
    zakazTableAdapter.Update(vidioDataSet4);
}
```

```
private void button3_Click(object sender, EventArgs e)
{
    za f = new za();
    f.Show();
    this.Close();
}
```

```
private void button5_Click(object sender, EventArgs e)
{
    Application.Exit();
}
```

```
private void za_Load_1(object sender, EventArgs e)
{
    // TODO: данная строка кода позволяет загрузить данные в таблицу
    "vidioDataSet4.Zakaz". При необходимости она может быть перемещена или удалена.
    this.zakazTableAdapter.Fill(this.vidioDataSet4.Zakaz);
}
```

```
}
```

Код Формы «Редактирования отзывов»

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace Vid
```

```
{
    public partial class kl : Form
    {
        public kl()
        {
            InitializeComponent();
        }
        class DB
```

```

    {
        public static VidioEntities connection = new VidioEntities();
    }
    private void button4_Click(object sender, EventArgs e)
    {
        za m = new za();
        m.Show();
        this.Close();
    }

    private void button6_Click(object sender, EventArgs e)
    {
        Form1 f = new Form1();
        f.Show();
        this.Close();
    }

    private void kl_Load(object sender, EventArgs e)
    {
        // TODO: данная строка кода позволяет загрузить данные в таблицу
"vidioDataSet3.Otzivi". При необходимости она может быть перемещена или удалена.
        this.otziviTableAdapter.Fill(this.vidioDataSet3.Otzivi);
    }

    private void button2_Click(object sender, EventArgs e)
    {
        otziviTableAdapter.Update(vidioDataSet3);
    }

    private void button1_Click(object sender, EventArgs e)
    {
        kl f = new kl();
        f.Show();
        this.Close();
    }

    private void button5_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }
}

```

Код Формы «Выбора услуг и оформления заказа»

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

```

```

namespace Vid
{

```

```

public partial class us : Form
{
    public us()
    {
        InitializeComponent();
    }
    string usluga = "";
    private void button1_Click(object sender, EventArgs e)
    {
        panel1.Visible = true;
        usluga = button1.Text;
        label6.Text = "Выбранная услуга: " + usluga;
    }

    private void button2_Click(object sender, EventArgs e)
    {
        panel1.Visible = true;
        usluga = button2.Text;
        label6.Text = "Выбранная услуга: " + usluga;
    }

    private void button3_Click(object sender, EventArgs e)
    {
        panel1.Visible = true;
        usluga = button3.Text;
        label6.Text = "Выбранная услуга: " + usluga;
    }

    private void button5_Click(object sender, EventArgs e)
    {
        panel1.Visible = false;
    }

    private void us_Load(object sender, EventArgs e)
    {
        Size = new Size(330, 250);
        panel1.Location=new Point(12,12);
    }

    private void button4_Click(object sender, EventArgs e)
    {
        if ((textBox1.Text != "") && (textBox2.Text != "") && (textBox3.Text != ""))
        {
            SqlConnection con = new SqlConnection(@"Data Source=DESKTOP-62J7H61\SQLEXPRESS;Initial Catalog=Vidio;Integrated Security=True");
            con.Open();
            SqlCommand cmd = new SqlCommand("insert into
            dbo.Zakaz(Usluga,Name,Tele,Adress)" +
            "values('" + usluga + "', '" + textBox1.Text + "', '" + textBox2.Text + "', '" + textBox3.Text
            + "')", con);
            cmd.ExecuteNonQuery();

            ot z = new ot();
            z.Show();
            this.Close();
        }
    }
}

```

```

        else MessageBox.Show("Заполните все поля");
    }

    private void button6_Click(object sender, EventArgs e)
    {
        Form1 f = new Form1();
        f.Show();
        this.Close();
    }

    private void button7_Click(object sender, EventArgs e)
    {
        Application.Exit();
    }
}

```

Код Форма «ОТЗЫВОВ»

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

```

```

namespace Vid

```

```

{
    public partial class ot : Form
    {
        public ot()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            if ((textBox1.Text != "") && (richTextBox1.Text != ""))
            {
                SqlConnection con = new SqlConnection(@"Data Source=DESKTOP-62J7H61\SQLEXPRESS;Initial Catalog=Vidio;Integrated Security=True");
                con.Open();
                SqlCommand cmd = new SqlCommand("insert into dbo.Otzivi(Name,Otziv)+"
                    + "values('" + textBox1.Text + "', '" + richTextBox1.Text + "')", con);
                cmd.ExecuteNonQuery();
                MessageBox.Show("Спасибо за заказ");
                Form1 f = new Form1();
                f.Show();
                this.Close();
            }
            else MessageBox.Show("Заполните все поля");
        }
    }
}

```

```

private void button5_Click(object sender, EventArgs e)

```

```

    {
        Application.Exit();
    }
}
}
Код связи с БД
namespace Vid
{
    using System;
    using System.Data.Entity;
    using System.Data.Entity.Infrastructure;

    public partial class VidioEntities : DbContext
    {
        public VidioEntities()
            : base("name=VidioEntities")
        {
        }

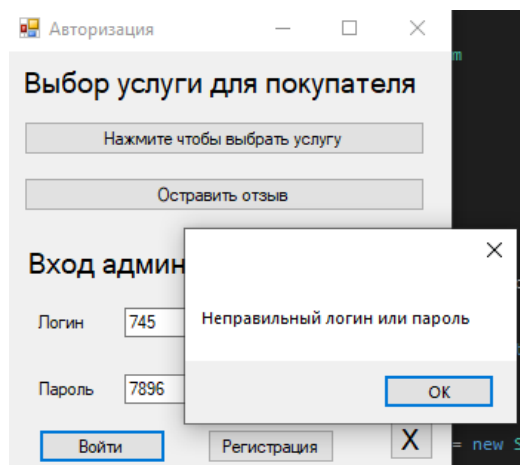
        protected override void OnModelCreating(DbModelBuilder modelBuilder)
        {
            throw new UnintentionalCodeFirstException();
        }

        public virtual DbSet<Otzivi> Otzivi { get; set; }
        public virtual DbSet<Zakaz> Zakaz { get; set; }
        public virtual DbSet<admin> admin { get; set; }
    }
}

```

## ТЕСТИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

1. Форма «Авторизация», проверка на неверный логин и пароль при нажатии на кнопку «Войти» или «Регистрация»



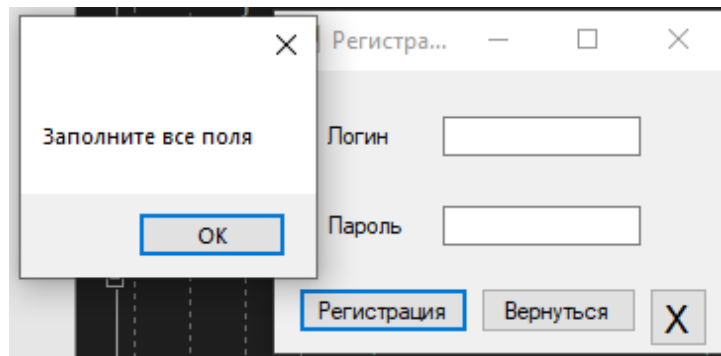
Форма «Авторизация», проверка на верный логин и пароль и открытие формы при нажатии на кнопку «Войти» или «Регистрация»

The screenshot shows a window titled 'Клиенты' (Clients) with a sub-header 'Редактирование клиентов' (Editing clients). It features a table with columns: 'id', 'Услуга' (Service), 'Имя' (Name), 'Телефон' (Phone), and 'Адрес' (Address). The table contains three rows of data. There are buttons for 'Редактирование отзывов' (Editing reviews) and 'Вернуться' (Return).

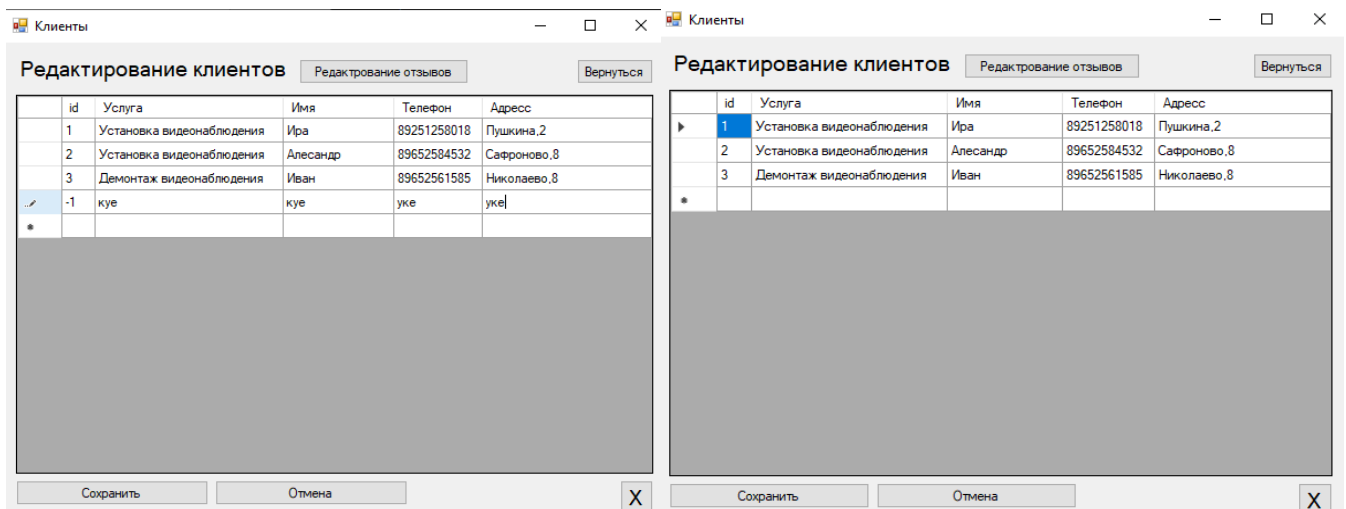
id	Услуга	Имя	Телефон	Адрес
1	Установка видеонаблюдения	Ира	89251258018	Пушкина,2
2	Установка видеонаблюдения	Александр	89652584532	Сафроново,8
3	Демонтаж видеонаблюдения	Иван	89652561585	Николаево,8

2. Форма «Регистрация нового администратора», регистрация (внесение данных в БД) при нажатии на кнопку «Регистрация»

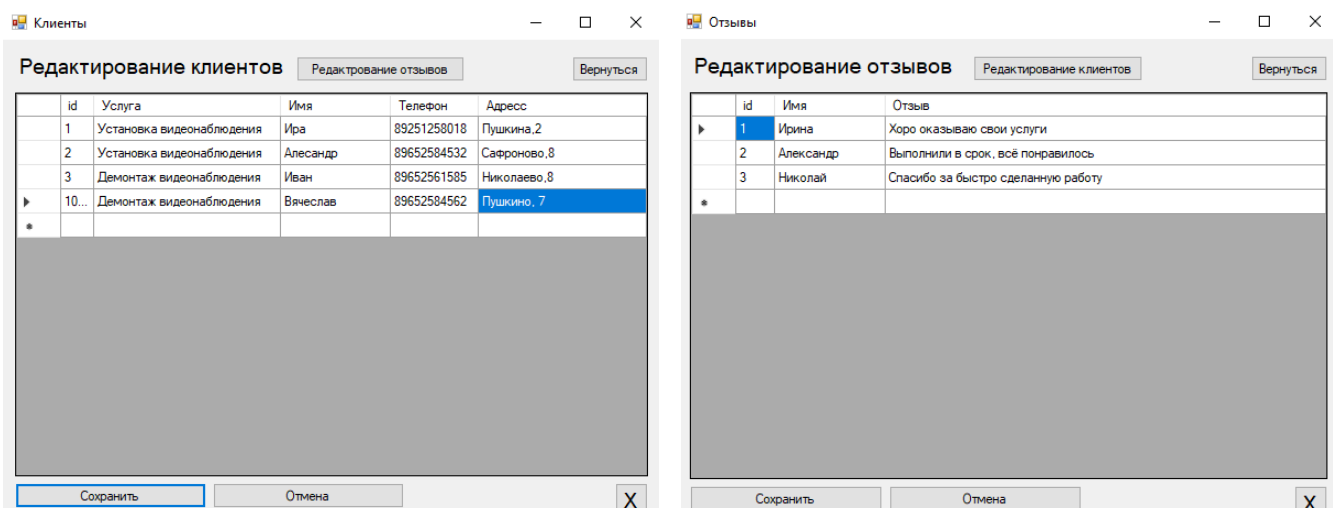
Форма «Регистрация нового администратора», ошибка при не заполненных полях при нажатии на кнопку «Регистрация»



3. Форма «Редактирования клиентов и заказов», при заполнении таблицы и нажатии на кнопку «Отмена» изменения не сохраняются

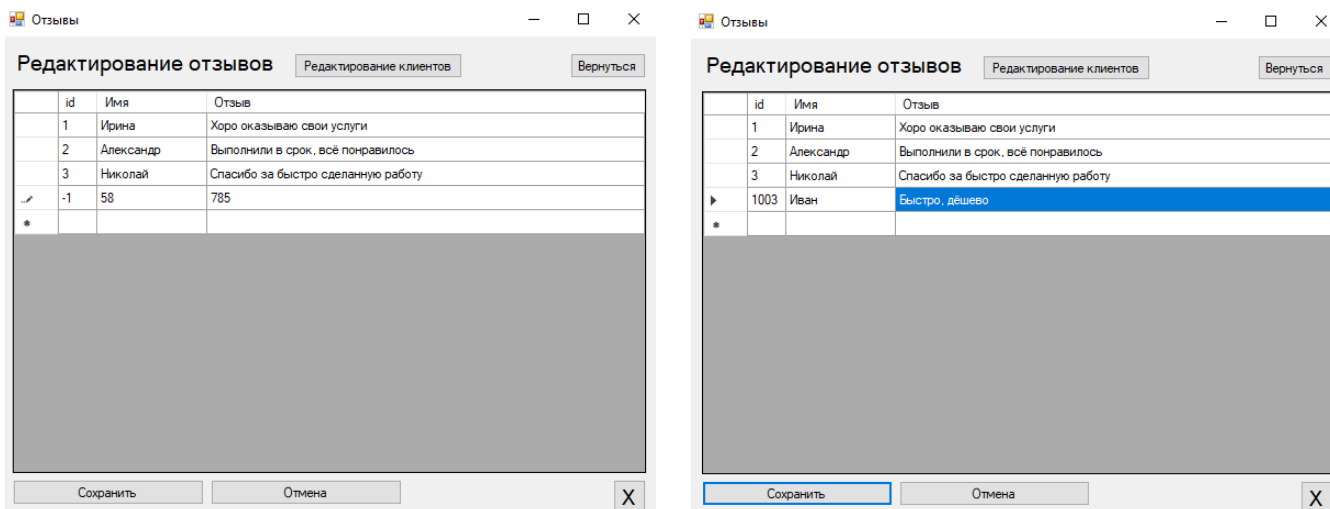


Форма «Редактирования клиентов и заказов», при заполнении таблицы и нажатии на кнопку «Сохранить» изменения сохраняются



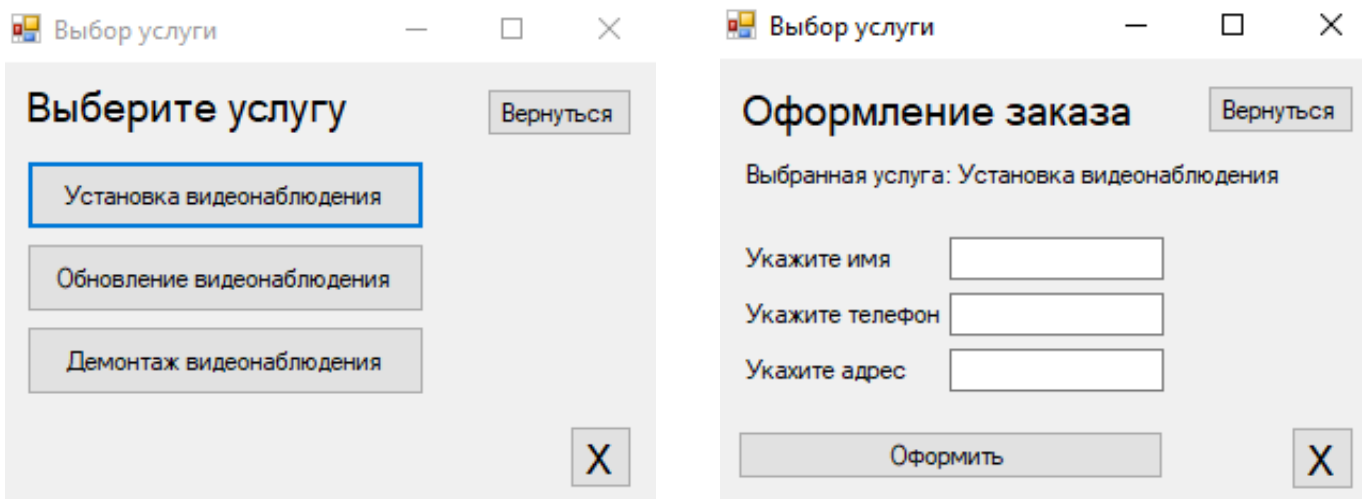


4. Форма «Редактирования отзывов», при заполнении таблицы и нажатии на кнопку «Отмена» изменения не сохраняются



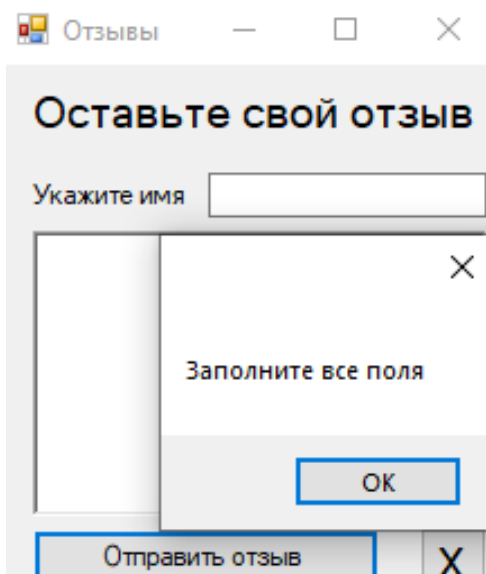
Форма «Редактирования отзывов», при заполнении таблицы и нажатии на кнопку «Сохранить» изменения сохраняются

5. Форма «Выбора услуг и оформления заказа», при нажатии на услугу открывается окно оформления заказа

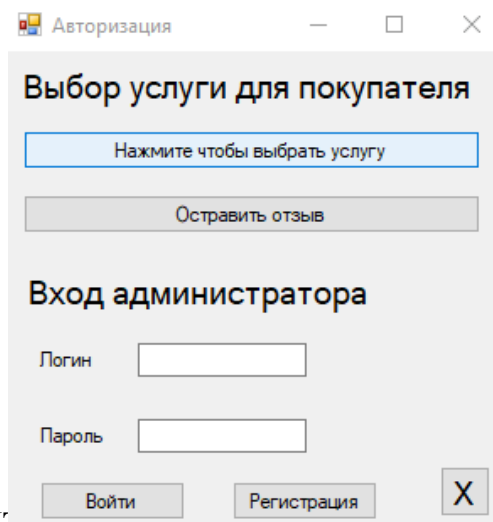


Форма «Выбора услуг и оформления заказа», при нажатии на кнопку «Оформить» заказ вносится в БД и открывается окно отзывов

6. Форма «Отзывы», при нажатии на кнопку «Отправить отзыв» при не заполненных полях появляется ошибка



Форма «Отзывы», при нажатии на кнопку «Отправить отзыв» при заполненных полях данные заносятся в БД и появляется окно «спасибо за заказ» на котором при нажатии «ОК» открывается форма «Авторизация»



ОПТИМИЗАЦИЯ КОДА

Код был оптимизирован в форме «Авторизация»

Изначальный вариант:

```
private void button1_Click(object sender, EventArgs e)
{
    if ((textBox1.Text != "") && (textBox2.Text != ""))
    {
        var email = textBox1.Text;
        var par = textBox2.Text;
        ...
    }
}

private void button2_Click(object sender, EventArgs e)
{
    if ((textBox1.Text != "") && (textBox2.Text != ""))
    {
        var email = textBox1.Text;
        var par = textBox2.Text;
        ...
    }
}
```

В этом случае можно вынести из кнопок присваивание и ничего не измениться, но уменьшит нагрузку и вес программы:

```
var email = textBox1.Text;
var par = textBox2.Text;

private void button1_Click(object sender, EventArgs e)
{
    if ((textBox1.Text != "") && (textBox2.Text != ""))
    {
        ...
    }
}

private void button2_Click(object sender, EventArgs e)
{
    if ((textBox1.Text != "") && (textBox2.Text != ""))
    {
        ...
    }
}
```

Код был оптимизирован в форме «Регистрация нового администратора»

Изначальный вариант:

```
class DB
{
    public static VidioEntities connection = new VidioEntities();
}
```

В этом случае можно удалить этот класс так как он не используется и ничего не изменится, но уменьшит нагрузку и вес программы:

...

## ОФОРМЛЕНИЕ ДОКУМЕНТАЦИИ ПОЛЬЗОВАТЕЛЯ

### *О программе*

Проект предназначен для более удобного заказа услуг по определённому адресу и возможности оставлять отзывы о проделанной работе по установке, обновлению и демонтажу видеонаблюдения. Также даёт возможность администратору просматривать и редактировать заказы и отзывы.

### *Форма авторизации*

На форме размещены кнопки выбора услуги и отзывов, поля для ввода логина и пароля, кнопки входа и регистрации. При нажатии на кнопку выбора услуги и отзывов открываются соответствующие формы. При нажатии на кнопку войти и регистрации происходит проверка данных, и открываются соответствующие формы.

### *Форма регистрации нового администратора*

На форме размещены поля для ввода логина и пароля, кнопку регистрации (только пользователь с ролью «администратор» может добавлять нового администратора)

### *Форма выбора услуги*

На форме размещены кнопки выбора услуги, при нажатии на которые открывается окно оформления заказа, где пишется выбранная услуга и поля для ввода, имени, телефона, адреса и кнопка оформления заказа с переходом на форму отзывов.

### *Форма отзывов*

На форме размещены поля для ввода имени и отзыва, и кнопка отправки отзыва.

### *Форма клиенты*

На форме расположено окно редактирования данных о клиентах и кнопки сохранить и отмена, а также кнопка для перехода на форму редактирования отзывов.

### *Форма отзывы*

На форме расположено окно редактирования отзывов и кнопки сохранить и отмена, а также кнопка для перехода на форму клиенты.

Доклад на тему:

**«Разработка программного обеспечения для автоматизации  
записи посетителей на спортивные занятия»**

## Место прохождения практики: Дворец спорта «Егорьевск»

Цель данного проекта является разработка программного обеспечения, чтобы любой человек из дома или работы записаться на спортивные занятия.

Первая задача это сделать окно с информацией о занятиях, чтобы можно было открыть и посмотреть какие занятия есть. И их описание, что данное секция(занятие) из себя представляет.

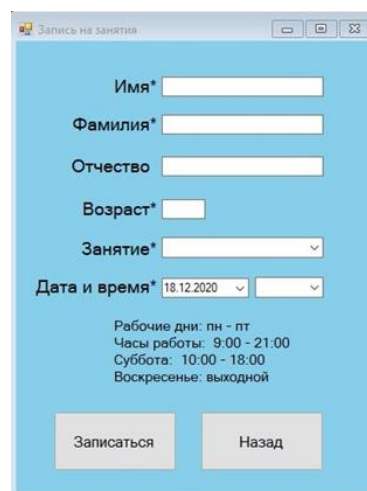
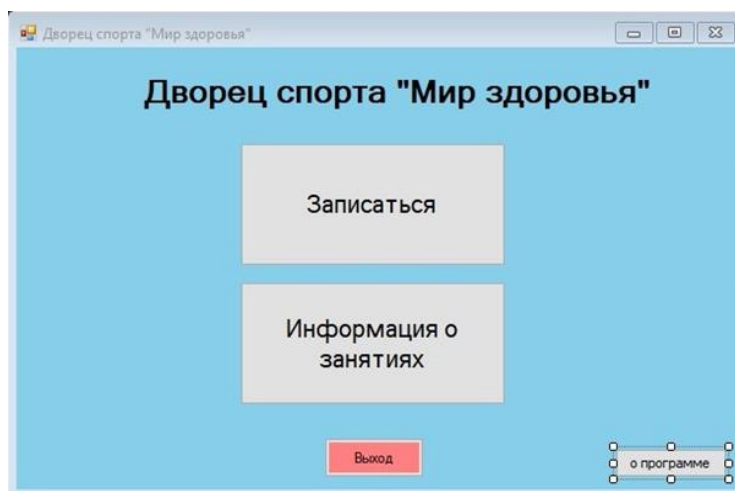
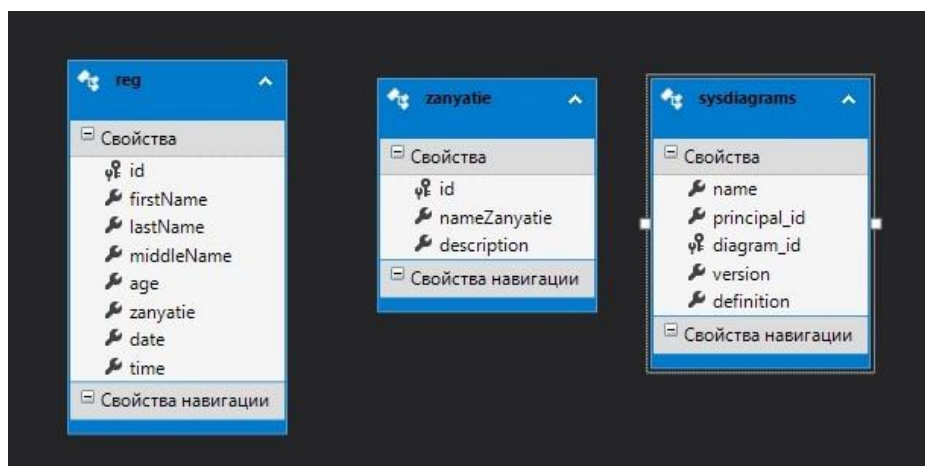
Вторая задача сделать форму с записью на конкретное занятие. В ней вы вводите имя, фамилию, отчество, если оно присутствует, возраст, занятие на которое хотите пойти, дальше дату, время. Внизу сразу можно посмотреть вообще когда работает в данном случае дворец спорта.

Третья задача вытекает из второй. После записи показывать информацию, которую вы заполнили.

Чтобы хранить информацию о занятиях требуется разработать базу данных. Для это использовал SQL и ADO.NET технологию подключения базы данных к проекту. Сделано две таблицы. В одной храниться занятия и информация о них. В другой находится информация о людях, которые записались на занятие.



**Гладков Илья –**  
студент группы Ип-73



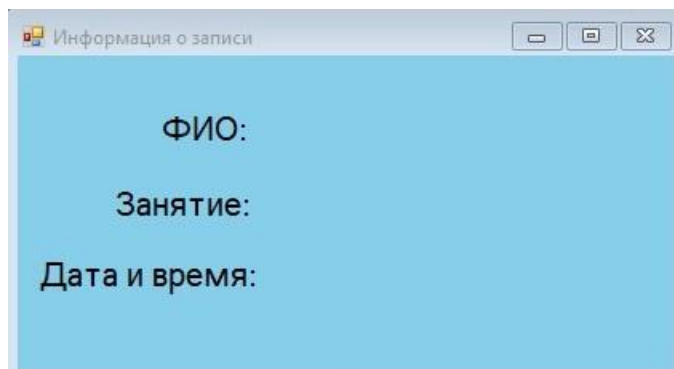
В проекте использовалась технология windows forms, с помощью её разрабатываются все окна и программируются. Проект написан на языке с#. Он выбран, с целью изучения.

На главном окне есть две большие кнопки. Верхняя записаться, нижняя информация о занятиях.

На окне с информацией почти во всё окно размещается элемент, который показывает информацию всех занятий и информацию о них и кнопку назад чтобы можно было вернуться не главную.

Форма записи состоит из разных элементов, таких как полей для ввода, поля выбора, и кнопок.

После того как вы внесли данные следует нажать на кнопку записаться, если не заполнили появиться ошибка, иначе появиться окно с данными, вашей фамилией, именем, отчеством, на какое занятие записались и дата с временем.



Доклад на тему:

### ***«Корпоративные сети – один из путей повышения эффективности работы предприятия»***

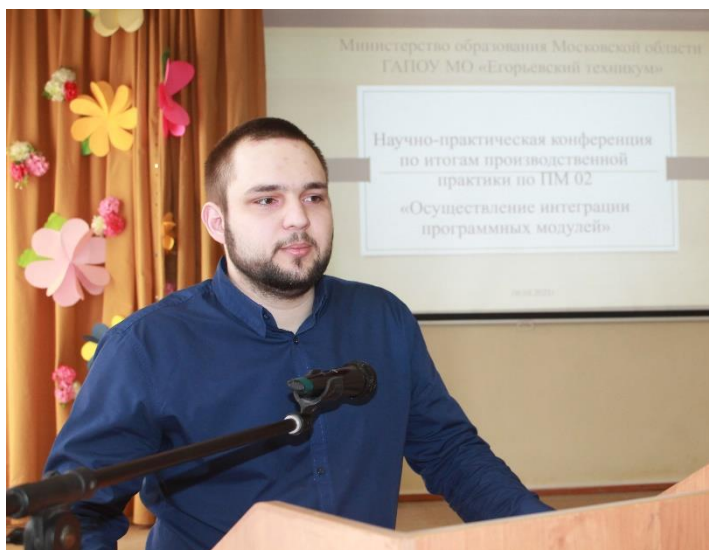
Корпоративная сеть - это сеть, главным назначением которой является обеспечение функционирования конкретного предприятия, владеющего данной сетью.

Пользователями корпоративной сети являются только сотрудники данного **предприятия**. В отличие от сетей операторов связи, корпоративные сети, в общем случае, не оказывают услуг другим организациям или пользователям.

В зависимости от масштаба предприятия, а также от сложности и многообразия решаемых задач различают сети отдела, сети кампуса и корпоративные сети (термин "корпоративные" в данной классификации приобретает узкое значение - сеть большого предприятия). Прежде чем обсуждать характерные особенности каждого из перечисленных типов сетей, остановимся на тех факторах, которые заставляют предприятия обзаводиться собственной компьютерной сетью.

**Что дает предприятию использование сетей**

Этот вопрос можно уточнить следующим образом:



**Пир Александр Александрович – преподаватель общепрофессиональных дисциплин**

- В каких случаях развертывание на предприятии вычислительных сетей предпочтительнее использования автономных компьютеров или многомашинных систем?

- Какие новые возможности появляются на предприятии с появлением вычислительной сети?

- И, наконец, всегда ли предприятию нужна сеть?

Если не вдаваться в подробности, то конечной целью использования компьютерных сетей на предприятии является повышение эффективности его работы, которое может выражаться, например, в увеличении прибыли. Действительно, если благодаря компьютеризации снизились затраты на производство уже существующего продукта, сократились сроки разработки новой модели или ускорилось обслуживание заказов потребителей - это означает, что данному предприятию в самом деле нужна была сеть.

Концептуальным **преимуществом сетей**, которое вытекает из их принадлежности к распределенным системам, перед автономно работающими компьютерами является их способность выполнять параллельные вычисления. За счет этого в системе с несколькими обрабатывающими узлами в принципе можно достичь **производительности**, превышающей максимально возможную на данный момент производительность любого отдельного, сколь угодно мощного, процессора. Распределенные системы потенциально имеют лучшее соотношение производительность/стоимость, чем централизованные системы.

Еще одно очевидное и важное достоинство распределенных систем - это их более высокая отказоустойчивость. Под отказоустойчивостью следует понимать способность системы выполнять свои функции (может быть, не в полном объеме) при отказах отдельных элементов аппаратуры и неполной доступности данных. Основой повышенной отказоустойчивости распределенных систем является избыточность. Избыточность обрабатывающих узлов (процессоров в многопроцессорных системах или компьютеров в сетях) позволяет при отказе одного узла переназначать приписанные ему задачи на другие узлы. С этой целью в распределенной системе могут быть предусмотрены процедуры динамической или статической реконфигурации. В вычислительных сетях некоторые наборы данных могут дублироваться на внешних запоминающих устройствах нескольких компьютеров сети, так что при отказе одного из них данные остаются доступными.

Использование территориально распределенных вычислительных систем больше соответствует распределенному характеру прикладных задач в некоторых предметных областях, таких как автоматизация технологических процессов, банковская деятельность и т. п. Во всех этих случаях имеются рассредоточенные по некоторой территории отдельные потребители информации - сотрудники, организации или технологические установки. Эти потребители автономно решают свои задачи, поэтому следовало бы предоставлять им собственные вычислительные средства, но в то же время, поскольку решаемые ими задачи логически тесно взаимосвязаны, их вычислительные средства должны быть объединены в общую систему. Оптимальным решением в такой ситуации является использование вычислительной сети.

Для пользователя распределенные системы дают еще и такие преимущества, как возможность совместного использования данных и устройств, а также возможность гибкого распределения работ по всей системе. Такое разделение дорогостоящих периферийных устройств - таких как дисковые массивы большой емкости, цветные принтеры, графопостроители, модемы, оптические диски - во многих случаях является основной причиной развертывания сети на предприятии. Пользователь современной вычислительной сети работает за своим компьютером, часто не отдавая себе отчета в том, что он пользуется данными другого мощного компьютера, находящегося за сотни километров от него. Он отправляет электронную почту через модем, подключенный к коммуникационному серверу, общему для нескольких отделов его предприятия. У пользователя создается впечатление, что эти ресурсы подключены непосредственно к его компьютеру или же "почти" подключены, так как для работы с ними нужны незначительные дополнительные действия по сравнению с использованием действительно собственных ресурсов.

В последнее время стал преобладать другой побудительный мотив развертывания сетей, гораздо более важный в современных условиях, чем экономия средств за счет разделения между сотрудниками корпорации дорогой аппаратуры или программ. Этим мотивом стало стремление обеспечить сотрудникам оперативный доступ к обширной корпоративной информации. В условиях жесткой конкурентной борьбы в любом секторе рынка выигрывает, в конечном счете,

та компания, сотрудники которой могут быстро и правильно ответить на любой вопрос клиента - о возможностях их продукции, об условиях ее применения, о решении различных проблем и т. п. На крупном предприятии даже хороший менеджер вряд ли знает все характеристики каждого из выпускаемых продуктов, тем более что их номенклатура может обновляться каждый квартал, если не месяц. Поэтому очень важно, чтобы менеджер имел возможность со своего компьютера, подключенного к корпоративной сети, скажем, в Магадане, передать вопрос клиента на сервер, расположенный в центральном отделении предприятия в Новосибирске, и оперативно получить ответ, удовлетворяющий клиента. В таком случае клиент не обратится в другую компанию, а будет пользоваться услугами данного менеджера и впредь.

Использование сети приводит к совершенствованию **коммуникаций** между сотрудниками предприятия, а также его клиентами и поставщиками. Сети снижают потребность предприятий в других формах передачи информации, таких как телефон или обычная почта. Зачастую именно возможность организации электронной почты является одной из причин развертывания на предприятии вычислительной сети. Все большее распространение получают новые технологии, которые позволяют передавать по сетевым каналам связи не только компьютерные данные, но и голосовую и видеoinформацию. Корпоративная сеть, которая интегрирует данные и мультимедийную информацию, может использоваться для организации аудио- и видеоконференций, кроме того, на ее основе может быть создана собственная внутренняя телефонная сеть.

#### **Преимущества, которые дает использование сетей**

1. Интегральное преимущество – повышение эффективности работы предприятия.
2. Способность выполнять параллельные вычисления, за счет чего может быть повышена производительность и отказоустойчивость.
3. Большее соответствие распределенному характеру некоторых прикладных задач.
4. Возможность совместного использования данных и устройств.
5. Возможность гибкого распределения работ по всей системе.
6. Оперативный доступ к обширной корпоративной информации.
7. Совершенствование коммуникаций.

#### **Проблемы**

1. Сложность разработки системного и прикладного программного обеспечения для распределенных систем.
2. Проблемы с производительностью и **надежностью** передачи данных по сети.
3. Проблема обеспечения безопасности.

Конечно, при использовании вычислительных сетей возникают и проблемы, связанные в основном с организацией эффективного взаимодействия отдельных частей распределенной системы.

Во-первых, это неполадки в программном обеспечении: операционных системах и приложениях. Программирование для распределенных систем принципиально отличается от программирования для централизованных систем. Так, сетевая операционная система, выполняя в общем случае все функции по управлению локальными ресурсами компьютера, сверх того решает многочисленные задачи, связанные с предоставлением сетевых услуг. Разработка сетевых приложений осложняется из-за необходимости организовать совместную работу их частей, выполняющихся на разных машинах. Массу хлопот доставляет и обеспечение совместимости программного обеспечения, устанавливаемого в узлах сети.

Во-вторых, много проблем связано с транспортировкой сообщений по каналам связи между компьютерами. Основные задачи здесь - обеспечение надежности (чтобы передаваемые данные не терялись и не искажались) и производительности (чтобы обмен данными происходил с приемлемыми задержками). В структуре общих затрат на вычислительную сеть расходы на решение "транспортных вопросов" составляют существенную часть, в то время как в централизованных системах эти проблемы полностью отсутствуют.

В-третьих, это вопросы, связанные с обеспечением безопасности, которые гораздо сложнее решать в вычислительной сети, чем в автономно работающем компьютере. В некоторых случаях, когда безопасность особенно важна, от использования сети лучше отказаться.

Можно приводить еще много "за" и "против", но главным доказательством эффективности использования сетей является бесспорный факт их повсеместного распространения. Сегодня трудно найти предприятие, на котором нет хотя бы односегментной сети персональных

компьютеров; все больше и больше появляется сетей с сотнями рабочих станций и десятками серверов, некоторые крупные организации обзаводятся частными глобальными сетями, объединяющими их филиалы, удаленные на тысячи километров. В каждом конкретном случае для создания сети были свои основания, но верно и общее утверждение: что-то в этих сетях все-таки есть.

### Сети отделов

Сети отделов - это сети, которые используются сравнительно небольшой группой сотрудников, работающих в одном отделе предприятия. Эти сотрудники решают некоторые общие задачи, например ведут бухгалтерский учет или занимаются маркетингом. Считается, что отдел может насчитывать до 100-150 сотрудников.

Главной целью сети отдела является **разделение локальных ресурсов**, таких как приложения, данные, лазерные принтеры и модемы. Обычно сети отделов имеют один или два файловых сервера, не более тридцати пользователей ([рис. 1](#)) и не разделяются на подсети. В этих сетях локализуется большая часть трафика предприятия. Сети отделов обычно создаются на основе какой-либо одной сетевой технологии - Ethernet, TokenRing. В такой сети чаще всего используется один или, максимум, два типа операционных систем. Небольшое количество пользователей позволяет применять в сетях отделов одноранговые сетевые ОС, например WindowsXP.

**Рис. 1.** Пример сети масштаба отдела



Задачи управления сетью на уровне отдела относительно просты: добавление новых пользователей, устранение простых отказов, инсталляция новых узлов и установка

новых версий программного обеспечения. Такой сетью может управлять сотрудник, посвящающий выполнению обязанностей администратора только часть своего времени. Чаще всего администратор сети отдела не имеет специальной подготовки, но является тем человеком в отделе, который лучше всех разбирается в компьютерах, и само собой получается так, что он и занимается администрированием сети.

Существует и другой тип сетей, близкий к сетям отделов, - сети рабочих групп. К таким сетям относят совсем небольшие сети, включающие до 10-20 компьютеров. Характеристики сетей рабочих групп практически не отличаются от описанных выше характеристик сетей отделов. Такие свойства, как простота сети и однородность, здесь проявляются в наибольшей степени, в то время как сети отделов могут приближаться в некоторых случаях к следующему по масштабу типу сетей - сетям кампусов.

### Сети кампусов

Сети кампусов получили свое название от английского слова campus - студенческий городок. Именно на территории университетских городков часто возникала необходимость в объединении нескольких мелких сетей в одну большую. Сейчас это название не связывают со



студенческими городками, а используют для обозначения сетей любых предприятий и организаций.

Сети кампусов (рис. 2) объединяют множество сетей различных отделов одного предприятия в пределах отдельного здания или одной территории, покрывающей площадь в несколько квадратных километров. При этом глобальные соединения в сетях кампусов не используются. Службы такой сети включают взаимодействие между сетями отделов, доступ к общим базам данных предприятия, доступ к общим факс-серверам, высокоскоростным модемам и высокоскоростным принтерам. В результате сотрудники каждого отдела предприятия получают доступ к некоторым файлам и ресурсам сетей других отделов. Сети кампусов обеспечивают доступ к корпоративным базам данных независимо от того, на каких типах компьютеров они располагаются.

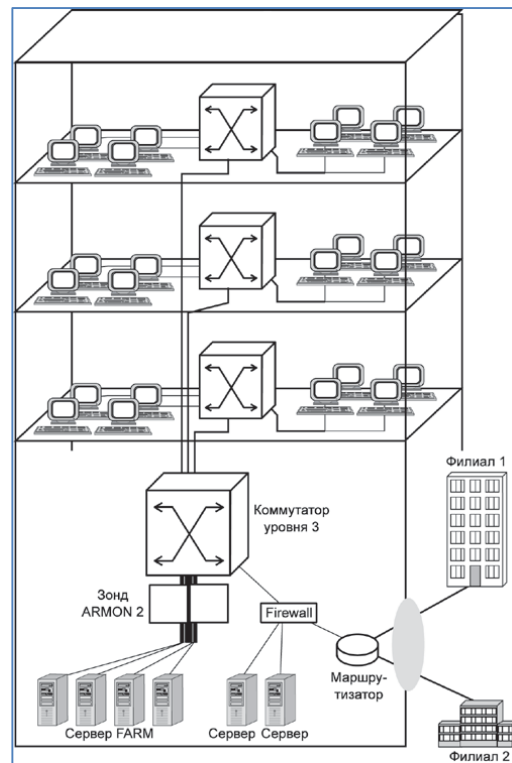


Рис. 2. Пример сети кампуса

Именно на уровне сети кампуса возникают проблемы интеграции неоднородного аппаратного и программного обеспечения. Типы компьютеров, сетевых операционных систем, сетевого аппаратного обеспечения в каждом отделе могут отличаться. Отсюда вытекают сложности управления сетями кампусов. Администраторы должны быть в этом случае более квалифицированными, а средства оперативного управления сетью - более эффективными.

### Сети масштаба предприятия

Корпоративные сети называют также сетями масштаба предприятия, что соответствует дословному переводу термина "enterprise-widenetworks", используемого в англоязычной литературе для обозначения этого типа сетей. Сети масштаба предприятия (корпоративные сети) объединяют большое количество компьютеров на всех территориях отдельного предприятия. Они могут быть сложно связаны и способны покрывать город, регион или даже континент. Число пользователей и компьютеров может измеряться тысячами, а число серверов - сотнями, расстояния между сетями отдельных территорий бывают такими, что приходится использовать глобальные связи (рис. 3). Для соединения удаленных локальных сетей и отдельных компьютеров в корпоративной сети применяются разнообразные телекоммуникационные средства, в том числе телефонные каналы, радиоканалы, спутниковая связь. Корпоративную сеть можно представить в виде "островков локальных сетей", плавающих в телекоммуникационной среде.

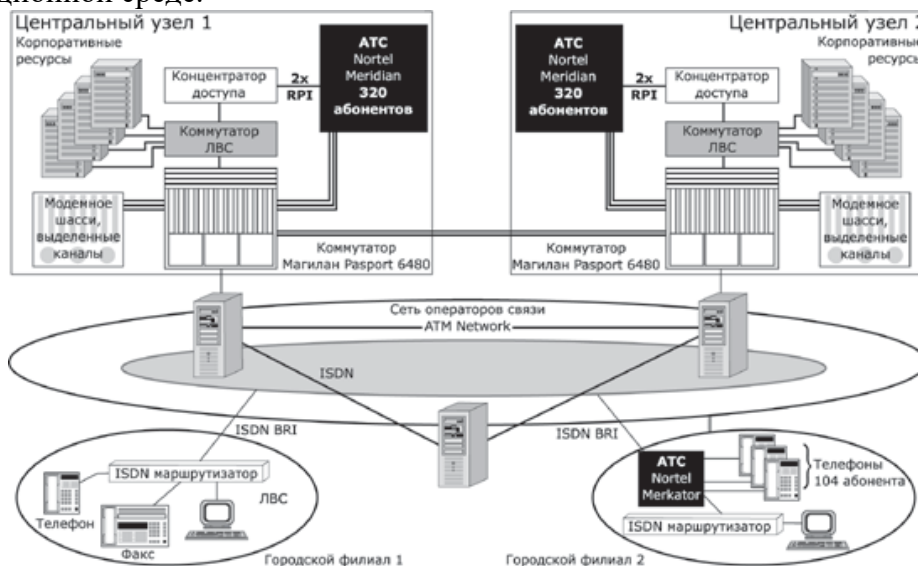


Рис. 3. Пример корпоративной сети

Непременным атрибутом такой сложной и крупномасштабной сети является высокая степень неоднородности (гетерогенности) - нельзя удовлетворить потребности тысяч пользователей с помощью однотипных программных и аппаратных средств. В корпоративной сети обязательно будут использоваться различные типы компьютеров - от мэйнфреймов до персоналок, несколько типов операционных систем и множество различных приложений. Неоднородные части корпоративной сети должны работать как единое целое, предоставляя пользователям по возможности удобный и простой доступ ко всем необходимым ресурсам.

Сети предприятий (корпоративные сети) объединяют большое количество компьютеров на всех территориях отдельного предприятия. Для корпоративной сети характерны:

- масштабность - тысячи пользовательских компьютеров, сотни серверов, огромные объемы хранимых и передаваемых по линиям связи данных, множество разнообразных приложений;
- высокая степень гетерогенности - различные типы компьютеров, коммуникационного оборудования, операционных систем и приложений;
- использование глобальных связей - сети филиалов соединяются с помощью телекоммуникационных средств, в том числе телефонных каналов, радиоканалов, спутниковой связи.

Появление корпоративных сетей - это хорошая иллюстрация известного постулата о переходе количества в качество. При объединении отдельных сетей крупного предприятия, имеющего филиалы в разных городах и даже странах, в единую сеть многие количественные характеристики объединенной сети переходят некоторый критический порог, за которым начинается новое качество. В этих условиях существующие методы и подходы к решению традиционных задач сетей меньших масштабов для корпоративных сетей оказались непригодными. На первый план вышли такие задачи и проблемы, которые в сетях рабочих групп, отделов и даже кампусов либо имели второстепенное значение, либо вообще не проявлялись. Примером может служить простейшая (для небольших сетей) задача - ведение учетных данных о пользователях сети.

Наиболее простой способ ее решения - поместить учетные данные каждого пользователя в локальную базу учетных данных каждого компьютера, к ресурсам которого пользователь должен иметь доступ. При попытке доступа эти данные извлекаются из локальной учетной базы, и на их основе предоставляется или не предоставляется доступ. В небольшой сети, состоящей из 5-10 компьютеров и примерно такого же количества пользователей, такой способ работает очень хорошо. Но если в сети насчитывается несколько тысяч пользователей, каждому из которых нужен доступ к нескольким десяткам серверов, то, очевидно, это решение становится крайне неэффективным. Администратор должен повторить несколько десятков раз (по числу серверов) операцию занесения учетных данных каждого пользователя. Сам пользователь также вынужден повторять процедуру логического входа каждый раз, когда ему нужен доступ к ресурсам нового сервера. Хорошее решение этой проблемы для крупной сети - использовать централизованную справочную службу, в базе данных которой хранятся учетные записи всех пользователей сети. Администратор один раз выполняет операцию занесения данных пользователя в эту базу, а пользователь один раз выполняет процедуру логического входа, причем не в отдельный сервер, а в сеть целиком.

При переходе от более простого типа сетей к более сложному - от сетей отдела к корпоративной сети - территория охвата увеличивается, поддерживать связи компьютеров становится все сложнее. По мере увеличения масштабов сети повышаются требования к ее надежности, производительности и функциональным возможностям. По сети циркулирует все возрастающее количество данных, и необходимо обеспечивать их безопасность и защищенность наряду с доступностью. Все это приводит к тому, что корпоративные сети строятся на основе наиболее мощного и разнообразного оборудования и программного обеспечения.

Доклад на тему:

## **«Автоматизированные информационные системы»**

На сегодняшний день любое предприятие, фирма, организация обладает своей организационной структурой. Эта структура многомерна и может быть расчленена на несколько взаимосвязанных и взаимозависимых подструктур, которые можно рассматривать как самостоятельные структуры: структура управления производством, кадровая структура, маркетинговая, финансово-экономическая, информационная структуры. Все они находятся в тесном взаимодействии и именно их совокупность и создаёт организационную структуру предприятия. Одно из важнейших мест в этой структуре занимает информационная система.

В принципе, любую систему управления можно представить как информационную систему с различными информационными потоками в виде документов, распоряжений, запросов, обращающихся внутри организации, исходящих или входящих из внешней среды.

В последние десятилетия резко увеличился объём информации в обществе вообще и информации, используемой на предприятии в частности. Это связано с растущими темпами развития науки и техники, появлением новых технологий, быстрой их сменяемостью. На рынках сырья и продукции сложились условия, требующие постоянного наблюдения за состоянием рынка, его изменениями, тенденциями его развития, необходимо уметь предвидеть дальнейшее развитие ситуации и быть готовым к смене стратегии, стиля деятельности, технологии производства для быстреего приспособления к новым внешним условиям.

Всё это ведёт к тому, что в современных условиях руководителям предприятий приходится иметь дело с таким большим количеством информации, она так быстро меняется, что её часто становится просто невозможно обработать «вручную». Кроме того, на больших предприятиях с большими оборотами продукции и численностью работников существует необходимость учёта и контроля большого объёма финансовой, производственной, кадровой, закупочно-сбытовой, маркетинговой информации.

В связи с этим появляется необходимость создания автоматизированных систем сбора, обработки, хранения информации. Они должны облегчить процесс работы с информацией, циркулирующей на предприятии.

Появление компьютерной техники позволяет создать подобные системы. На современных предприятиях практически вся работа с информацией автоматизирована, существуют специальные программы, позволяющие вести на компьютере бухгалтерский учёт, документооборот, маркетинговые исследования, проводить прогнозирование и стратегическое планирование, а также многое другое. Но кроме автоматизации актуальным остаётся вопрос о грамотном построении структуры информационной системы, оптимизации информационных потоков, отсеивания ненужной информации, упрощения поиска и получения необходимой.

Наличие хорошо отлаженной автоматизированной информационной системы на предприятии значительно упрощает процесс управления предприятием. Она позволяет вовремя собрать, отсортировать, обработать необходимую информацию и принять верное решение. Иногда, не вовремя принятое решение, из-за недостатка или несвоевременного поступления информации может привести к гибели предприятия. Поэтому необходимо уделять большое внимание созданию и поддержанию эффективного функционирования информационной системы предприятия.



**Кирилина Ирина Анатольевна – преподаватель общепрофессиональных дисциплин**

В истории создания автоматизированных информационных систем относительно независимо развивались два направления:

1. разработка автоматизированных информационных систем (АИС) как автоматизированных систем управления (АСУ);
2. разработка автоматизированных систем научно-технической информации (АСНТИ).

Работы по их созданию начались практически одновременно в 60-е гг. Первое направление - разработка АИС и АСУ - было инициировано научно-техническим прогрессом и возникшими в связи с этим проблемами организационного управления (рост количества информации, трудности с её обработкой «вручную»).

Зарубежная практика шла по пути разработки отдельных программных процедур, например, для бухгалтерии, учета материальных ценностей, и основные работы проводились в направлении исследования и совершенствования возможностей вычислительной техники, разработки средств, обеспечивающих наиболее рациональную организацию информационных массивов, удобный для пользователя интерфейс, наращивание памяти ЭВМ.

В нашей стране проблема обеспечения информацией управленческих работников была поставлена сразу системно. Была разработана классификация АСУ, в которой прежде всего выделялись АСУ разных уровней системы управления - для уровня предприятий и организаций, отраслевые, республиканские и региональные и общегосударственная автоматизированная система Аналогично на уровне предприятий, и особенно создаваемых в 70-е гг. научно-производственных объединений (НПО), в структуре АСУП (или интегрированных АСУ объединений) выделялись уровни (страты) - АСУ объединения, АСУ предприятий и организаций (научно-исследовательских институтов, конструкторских бюро), входящих в НПО, АСУ производств, комплексов цехов, АСУ цехов и участков. На уровне цехов и участков АСУ вначале разделялись на АСУ технологическими процессами, АСУ технической и технологической подготовки производства, АСУ организацией производства.

Работы по созданию централизованных общегосударственных АСУ и АСНТИ были приостановлены в связи с преобразованиями 19-91 гг. Однако, при переходе к рыночной экономике, к правовому государству возрастает роль еще одного важного вида информации - нормативно-правовой и нормативно-методической, регламентирующей деятельность предприятий при предоставлении им большей самостоятельности и сокращении организационно-распорядительной документации (текущих приказов и распоряжений, ревизирующих командно-административные методы управления).

В дальнейшем, по мере развития предприятий и их АСУ, особенно в условиях предоставления большей самостоятельности производствам и цехам и перераспределению управленческих функций между администрацией предприятия и руководителями производств и цехов, также стало более удобным представлять структуру АСУ в виде многоуровневой, стратифицированной. Разделение АСУ на функциональную и обеспечивающую части, а последней - на информационное обеспечение, техническое, организационное, программное и другие виды обеспечения - позволило привлечь для уточнения соответствующих видов обеспечения специалистов в этих областях. Такой подход к организации разработок АСУ помог справиться со сложностью системы и ускорить разработку АСУ путем параллельного проведения работ по анализу и выбору структуры отдельных видов обеспечения. Однако, если разрабатывать отдельные проекты, то после разработки возникает достаточно сложная задача их согласования, взаимоувязки принятых структур этих видов обеспечения, критериев, учитываемых при их разработке и. Поэтому на определенном этапе развития работ по созданию АСУ был даже сформулирован специальный принцип - единства информационного обеспечения, технического и программного, как основных видов обеспечения.

В настоящее время существует огромное количество готовых программных продуктов. Поэтому, нет необходимости при создании на предприятии автоматизированной системы заниматься самостоятельной разработкой программного обеспечения.







## **Решение научно-практической конференции**

В результате работы научно-практической конференции на тему: «Профессионализм-основа успешной карьеры» по специальности 09.02.07 Информационные системы и программирование участники конференции решили:

1. Проведение систематической работы по корректировке содержания учебных рабочих программ теоретического и производственного обучения с учетом требований рынка труда;
2. Участие работодателей в работе государственных аттестационных комиссий при проведении итоговой государственной аттестации;
3. Внедрение в учебный процесс компьютерных и информационных технологий с целью подготовки специалиста, отвечающего современным требованиям работодателя;
4. Расширение тематики курсовых и дипломных проектов связанных с практическим применением;
5. Предоставление студентам для ознакомления новых программ применяемых в промышленном производстве;
6. Организация стажировки педагогических работников техникума для ознакомления с новейшими программами современных на предприятиях;
7. Ежегодное проведение научно-практической конференции по итогам производственной практики;
8. Регулярное проведение семинарских занятий с целью обобщения и систематизации знаний, анализа учебно-производственной деятельности студентов;
9. Издание по итогам научно-практической конференции 2021 года сборника докладов с размещением его на сайте техникума.