Содержание

[Введение 2](#_Toc117690921)

[1. Разработка системного проекта 3](#_Toc117690922)

[1.1 Структура оптимизации работы. 3](#_Toc117690923)

[2. Анализ предметной области 6](#_Toc117690924)

[3. Ресурсы по созданию базы данных и пользовательского интерфейса 7](#_Toc117690925)

[3.1 Ресурсы по созданию базы данных 7](#_Toc117690926)

[4. Реализация системного проекта 9](#_Toc117690927)

[4.1 Детальное проектирование 9](#_Toc117690928)

[5. Руководство пользователя 16](#_Toc117690929)

[5.1 Руководство пользователя 16](#_Toc117690930)

[Заключение 22](#_Toc117690931)

[Библиографический список 24](#_Toc117690932)

# Введение

Автоматизированное рабочее место (АРМ) - комплекс средств вычислительной техники и программного обеспечения, располагающийся непосредственно на рабочем месте сотрудника и предназначенный для автоматизации его работы в рамках специальности.

Автоматизация рабочего места представляет собой организацию места пользователя-специалиста той или иной профессии, оборудование средствами, необходимыми для автоматизации выполнения им определенных функций. Такими средствами, как правило, является ПК, дополняемый по мере необходимости другими вспомогательными электронными устройствами. На данный момент актуально освоение принципов построения и эффективного применения соответствующих технологий и программных продуктов: систем управления базами данных, средств администрирования и защиты бах данных.

В данной курсовой работе требуется организовать работу по слежению(учету) банкоматов, вывод необходимой информации о банкоматах, разработать современный дизайн для информационной системы.

Целью курсовой работы является создание информационной системы для организации слежения(учета) банкоматов.

Для реализации поставленной цели рассматриваются следующие задачи:

* изучить предметную область;
* выбрать СУБД;
* построить информационно-логическую модель;
* реализовать даталогическую модель с использованием СУБД;
* создать пользовательский интерфейс.

# Разработка системного проекта

# 1.1 Структура оптимизации работы.

АРМ состоит из технических и программных средств вычислительной техники, а также необходимой методической документации, позволяющей пользователю эффективно взаимодействовать с данными средствами.

В состав технических средств входит как непосредственно ЭВМ, на базе которой реализуется данное АРМ, так и периферийные устройства, и иные технические средства, набор которых может быть различен в зависимости от поставленных задач. Основными компонентами ЭВМ являются: центральный микропроцессор, системная шина, оперативная память, устройства ввода-вывода, накопители информации. Также к ЭВМ могут подключаться: печатающие устройства (принтеры, плоттеры), коммуникационное оборудование (модемы), устройства ввода изображений (сканеры) (Рисунок 1).

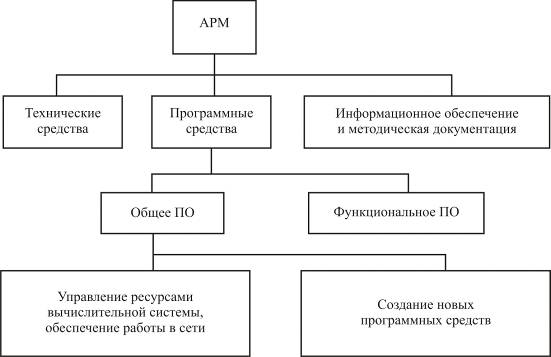


Рисунок 1. Схема автоматизированного рабочего места

Информационное обеспечение и методическая документация также играют важную роль в эффективном функционировании АРМ.

Информационное обеспечение означает постоянную информационную поддержку каждого отдельно взятого автоматизированного рабочего места. Функционирование современных АРМ невозможно без снабжения своевременной, достоверной и качественной информацией.

Методическая документация представляет собой комплекс документов, касающихся порядка функционирования, данного АРМ, и, как правило, включает в себя состав входных и выходных документов, инструкционные карты, должностные инструкции и другие документы. Создание продуманного, несложного в освоении комплекса методической документации особенно важно, когда в организации впервые внедряется система автоматизированных рабочих мест. В этом случае необходимо подробно объяснить сотрудникам порядок работы с новым для них оборудованием, пояснить все положительные стороны его использования. При необходимости следует организовать для работников посещение курсов повышения квалификации по работе с вычислительной техникой. Необходимо сделать все возможное, чтобы при внедрении в организации современных технологий обработки информации сотрудники не считали появившиеся технические средства помехой их обычной работе, а поняли всю выгоду и вес преимущества их использования.

Используемые в процессе построения и эксплуатации АРМ программные средства подразделяются на общее и функциональное программное обеспечение.

Общее программное обеспечение (ПО) обеспечивает функционирование вычислительной техники, разработку и подключение новых программ. В него входят операционные системы, системы программирования и обслуживающие программы (например, антивирусные, программные средства защиты информации). Еще одним важным элементом общесистемного программного обеспечения являются средства написания и отладки собственных программ пользователя. К ним относятся разнообразные редакторы, позволяющие создавать программы на различных языках программирования. Подобные программные средства могут понадобиться в том случае, если для работы необходимо создание собственных программ, аналогов которых нет на рынке программного обеспечения.

* 1. Функциональное программное обеспечение

Функциональное программное обеспечение (ФПО) определяет профессиональную ориентацию АРМ. Именно здесь реализуется направленность на конкретного специалиста, обеспечивается решение задач определенных предметных областей. Именно от состава функционального ПО зависит специализация конкретного АРМ. Так как ФПО в конечном счете определяет область применения АРМ и состав решаемых пользователем задач, то оно должно разрабатываться на основе программных средств диалоговых систем, предназначенных для выполнения функций со схожими процедурами обработки информации.

В зависимости от назначения АРМ в состав ФПО могут входить различные программы. Рассмотрим основные группы программных средств, используемых для автоматизации наиболее часто встречающихся функций, выполняемых сотрудниками. К подобным программным средствам относят следующие: системы подготовки текстовых документов (текстовые редакторы и настольные издательские системы); системы обработки финансово-экономической информации (табличные редакторы и другие подобные программы); системы управления базами данных; системы подготовки презентаций; системы управления проектами; личные информационные системы (органайзеры); Web‑браузеры; программы для работы с электронной почтой; экспертные системы; системы проектирования и совершенствования систем управления; системы обработки изображений документов; системы оптического распознавания символов; системы управления документами и организации электронного документооборота.

# Анализ предметной области

Актуальность данной темы заключается в том, что на сегодняшний день рабочее место практически любого человека является автоматизированным. Автоматизированное рабочее место является элементом системы управления, оснащенной средствами, с помощью которых обеспечивается участие человека в реализации функций автоматизации.

Сотрудник по слежению банкоматов сталкивается с необходимостью работать с большим количеством данных. В обязанности входит:

- подключение банкоматов из базы в список для отслеживания

- отслеживать на карте города

- отслеживать актуальность информации о банкоматах

Целью работы является создание программного продукта для автоматизированного рабочего место с использованием базы данных, с возможностью подключения и удаления объектов слежения.

Программный продукт, который будет создан, должен реализовывать следующие возможности:

- подключение и удаление объекта слежения

- поиск по ID объекта

- сортировка по месторасположению

# Ресурсы по созданию базы данных и пользовательского интерфейса

* 1. Выбор системы управления базами данных (СУБД)

Криптографические операции включают в себя шифрование столбцов, и эти операции теперь можно выполнять с помощью Transact-SQL, они не требуют перемещения данных из базы данных. Внутри защищенных анклавов поддерживаются все полнофункциональные вычисления, включая сопоставления и сравнения диапазонов, что значительно расширяет возможности их применения.

Технология Always Encrypted с защищенными анклавами доступна в Windows Server 2019(рисунок 4).

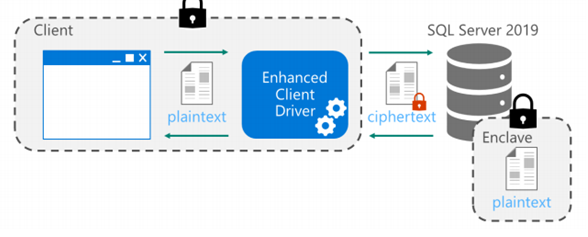


Рисунок 2. Технология Always Encrypted с защищенными анклавами доступна в Windows Server 2019.

1. Выбор ОС и контейнеров;

SQL Server 2019 отличается гибкостью в отношении выбора платформы, языка программирования и средства доставки.

* 1. Поддержка Red Hat Enterprise Linux, SUSE Linux Enterprise Server, Ubuntu и Windows;
  2. Контейнеры Docker для Linux и Windows. Установка со встроенной поддержкой инструментов Linux: Yum lnstall, Apt-Get и Zypper;
  3. Один и тот же уровень абстракции с SQL Server на Linux;
  4. Возможность использования R, Python и Java при работе с T-SQL. Теперь расширение языка Java доступно для выполнения кода Java в SQL Server;

1. Интеллектуальный анализ данных;

Развитие SQL Server пошло по пути интеграции с другими аналитическими платформами, в частности Spark, которая включена теперь в поставку SQL Server.

Spark является очень популярным инструментом для машинного обучения, для продвинутой аналитики, имеет эффективную in memory машину. In memory машина – это означает систему, предназначенную для использования преимуществ большего объема памяти, доступного для современных баз данных. База данных в памяти по своей природе может быть реляционной или нереляционной. Предполагается, что производительность системы баз данных, находящихся в памяти, в основном повышается за счет ускорения (достигающего нескольких порядков) доступа к данным, находящимся в памяти, по сравнению с данными, которые находятся даже в самой быстрой дисковой подсистеме по своей природе. Многие рабочие нагрузки SQL Server, однако, позволяют разместить весь рабочий набор в доступной памяти. Многие системы баз данных в памяти могут сохранять данные на диск и не всегда могут разместить весь набор данных в доступной памяти. И все это интегрировано с SQL, который очень эффективен для визуализации аналитики.

Правильный анализ и эффективное представление результатов напрямую влияет на эффективность анализа данных и возможность принимать на их основе управленческие решения.

Именно по этим перечисленным преимуществам база данных «Учет банкоматов» будет разработана в СУБД Microsoft SQL Server.

# 3.2 Ресурсы по созданию базы данных

За последние несколько лет вырос уровень потребительских качеств систем управления базами данных (СУБД): разнообразие поддерживаемых функций, удобный для пользователя интерфейс, сопряжение с программными продуктами, в частности с другими СУБД, возможности для работы в сети и т.д. СУБД позволяет сводить воедино информацию из самых разных источников (электронные таблицы, другие базы данных) и помогает быстро найти необходимую информацию, донести ее до окружающих с помощью отчетов, графиков или таблиц.

СУБД SQL Management Studio является системой управления реляционной базой данных, включающей все необходимые инструментальные средства для создания локальной базы данных, общей базы данных в локальной сети с файловым сервером или создания приложения пользователя, работающего с базой данных на SQL- сервере. Microsoft SQL Server входит в состав MS Office, что делает его интерфейс знакомым и привычным, а, следовательно, облегчает работу.

Для SQL Management Studio характерны следующие возможности:

* высокая степень производительности и отказоустойчивости;
* многопользовательский СУБД и работающий по принципу «клиент-сервер»;
* тесная интеграция с операционной системой Windows;
* поддержка удаленных подключений;
* поддержка популярных типов данных, а также возможность создания триггеров и хранимых процедур;
* встроенная поддержка ролей пользователей;
* расширенная функция резервного копирования баз данных;
* высокая степень защищенности;
* каждый выпуск включает в себя несколько специализированных редакций

SQL Server имеет характерный для всех приложений Microsoft Windows удобный графический интерфейс, ориентированный на комфортную работу пользователя. Для работы с таблицами базы данных и другими объектами SQL Server предоставляет многочисленные команды меню и контекстно-зависимые панели инструментов. Поскольку интерфейс приложений Microsoft Office унифицирован, пользователю требуется меньше времени на освоение приложения

Для разработки базы данных выбрана СУБД SQL Management Studio 2018.

3.2 Ресурсы по созданию пользовательского интерфейса

Интегрированная среда разработки (IDE) — это многофункциональная программа, которая поддерживает многие аспекты разработки программного обеспечения. Интегрированная среда разработки Visual Studio — это стартовая площадка для написания, отладки и сборки кода, а также последующей публикации приложений. Помимо стандартного редактора и отладчика, которые есть в большинстве сред IDE, Visual Studio включает в себя компиляторы, средства авто завершения кода, графические конструкторы и многие другие функции для улучшения процесса разработки.

Для разработки пользовательского интерфейса была выбрана Интегрированная среда разработки Visual Studio 2019.

3.3 Выбор языка программирования

Почти без раздумий был выбран C#, поскольку это один из самых популярных и достаточно нестрог к начинающим разработчикам. C# — пятый по популярности язык программирования в мире. Его используют банки, диджитал-агентства, провайдеры связи и крупные IT-компании. Вот что пишут на C# российские и зарубежные корпорации:

* Microsoft — приложения для Windows и Xbox.
* Tesla — корпоративные веб-сервисы и программы.
* Stack Overflow — серверную логику сайтов.
* Сбербанк — 3D-графику и программы виртуальной реальности.
* Ozon — складские и логистические системы.
* «Яндекс» — приложения для автоматизации продаж.

Поскольку C# — это язык от Microsoft, на нём удобно писать настольные приложения для Windows. Для этого разработчики придумали специальную платформу — называется .NET Framework. Она умеет переводить код из понятного человеку в понятный процессору — то есть **компилировать**.

Преимущество платформы в том, что она может исполнить любую C#-команду на любом процессоре — а на Windows работает не меньше тысячи разных моделей. Если бы не .NET, пришлось бы компилировать код под каждое железо отдельно. А ещё .NET поддерживает много плагинов, библиотек и шаблонов для наглядной разработки интерфейсов — это фактически целый мир и экосистема для программиста на Windows.

3.4 Среда разработки

Интегрированная среда разработки Visual Studio является творческой стартовой площадкой, которую можно использовать для редактирования, отладки и сборки кода, а также для публикации приложения. В дополнение к стандартному редактору и отладчику, предоставляемых большинством интегрированных сред разработки, Visual Studio включает компиляторы, средства завершения кода, графические конструкторы и многие другие функции для улучшения процесса разработки программного обеспечения.

Среда Visual Studio доступна для Windows и Mac. Функции Visual Studio для Mac во многом аналогичны возможностям Visual Studio для Windows и оптимизированы для разработки кроссплатформенных и мобильных приложений. Эта статья посвящена версии Visual Studio для Windows. Существует три выпуска Visual Studio: Community, Professional и Enterprise.

Популярные средства повышения производительности:

* Волнистые линии и быстрые действия
* Очистка кода
* Рефакторинг
* IntelliSense
* Поиск в Visual Studio
* Live Share
* Иерархия вызовов
* CodeLens
* Показать определение
* Перейти к определению
* Горячая перезагрузка

Средой разработки для польщовательского интерфейса была выбрана Microsoft Visual Studio, в основном из-за ее простоты в освоении и доступности для бесплатного использование без ограничений функционала.

3.5 Преимущества C#

**Независимость от железа.**Программисту не надо адаптировать программу под разные платформы и системы — за него это делает виртуальная машина, вшитая в .NET Framework. В итоге один и тот же код можно запускать на любых устройствах — смартфонах, компьютерах, серверах, банкоматах и даже умных часах.

**Отличная совместимость с Windows.**  Не зря же язык разработали именно в Microsoft. Так же как Swift идеально подходит для программирования под экосистему Apple, C# прекрасно вписывается в экосистему Windows.

**Управление памятью.**  Чтобы программа работала стабильно, её надо иногда чистить от ненужных объектов, ссылок, кэша и прочего мусора. В C# это происходит автоматически — разработчику не надо следить за расходом памяти, бороться с её утечками или удалять мёртвые куски кода.

**Строгая типизация.** Когда вы объявляете переменную в C#, надо сначала указать, что в ней лежит — строка, число или массив. Так разрабатывать чуть дольше, зато ваш код работает предсказуемо — числа взаимодействуют с числами, строки со строками и так далее. В языках со слабой типизацией свободы и драйва больше, но есть шанс пропустить ошибку, которая всплывёт в готовой программе.

**Большое сообщество.** На С# пишут более миллиона программистов по всему миру. В соцсетях полно чатов и сообществ «шарпистов», где можно задать вопрос, обсудить сложную тему или найти готовое решение. В теории можно даже найти ментора, который поделится знаниями и поможет быстрее освоить язык.

**Синтаксический сахар.**В С# есть много способов сократить код, не нарушая логику программы. Программисты называют такие приёмы «синтаксическим сахаром» — они помогают сделать код проще, понятнее и в целом симпатичнее. Сравните, например, как выглядит сложение чисел с «сахаром» и без.

3.6 Преимущества WPF

Плюсы:

* При ошибке в работе дизайнера ничего фатального произойти не может, потому что внешний вид приложения описывается как xaml
* Возможность настроить внешний вид стандартных элементов управления практически как угодно, плюс разработать свои нестандартные элементы управления
* Проблемы при масштабировании экрана в новых Windows возникают гораздо реже, чем при использовании Windows Forms
* Привязка данных гибко настраивается
* Поддержка анимации, звука, 3D графики

Минусы:

* Писать xaml гораздо дольше, чем пользоваться дизайнером Windows Forms. Кроме того, декларативный код и большей концентрации требует при написании, и отлаживать его практически невозможно. Ну и в целом архитектура технологии не располагает к быстрой разработке, сложнее автоматизировать рутину.
* Непредсказуемое поведение при изменении темы Windows. Теоретически, WPF подстраивается под тему Windows, но на практике её элементы управления выглядят по-разному в зависимости от темы Windows, но всё равно не нативно. Будет точнее сказать, что она пытается это делать, но делает плохо.
* Плохая интеграция с базовыми библиотеками .NET Framework. Такое ощущение, что WPF писали в целом далёкие от основного стека .NET люди
* Как следствие, отсутствует поддержка не визуальных компонентов и детерминированного уничтожения. Компонентная модель не то чтобы совсем не поддерживается, но как-то уж очень выборочно.
* Стандартные элементы управления содержат ряд странных недоработок (например, полосы прокрутки без возможности задать минимальный размер ползунка, а также их некрасивое расположение в таблицах данных).
* Неудобная система задания расположения элементов управления через явные номера строк и столбцов в самом распространённом контейнере. При изменении во внешнем виде приложения нужно их исправлять во всех элементах управления.
* Постоянно что-нибудь масштабируется, даже если явно просишь это не делать, и в результате размывается - шрифты, растровые картинки. Причём ещё и не всегда, это зависит от монитора, видеокарты и темы Windows.
* Использование рекомендуемого Microsoft паттерна MVVM приводит к раздуванию объёма исполняемого кода.
* Отсутствует диалог выбора директории из коробки
* Отсутствует поддержка MDI из коробки. Поскольку это базовая функция практически в любом приложении, приходится либо делать самостоятельно, либо искать сторонние решения
* Высокое потребление системных ресурсов
* Иногда начинает глючить сам графический движок и рисовать хаос из пикселей вместо того, что запрограммировано

# 4. Реализация системного проекта

# 4.1 Детальное проектирование

Для создания информационной модели нужно определить сущности, которые будут удовлетворять всем требования к базе данных:

1. ATM
2. Sector
3. Sector\_GPS
4. Street
5. Group\_Sector
6. Group\_
7. Group\_Person
8. Person
9. City

Для каждой сущности определим атрибуты, которые будут храниться в базе данных.

Таблица 1. Сущность “ATM”

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Разрешить значения Null | Первичный ключ |
| id\_ATM | int | - | + |
| balance | money | + | - |
| isConnected | char(1) | + | - |

Таблица 2. Сущность “Sector”

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Разрешить значения Null | Первичный ключ |
| id\_sector | int | - | + |

Таблица 3. Сущность “Sector\_GPS”

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Разрешить значения Null | Первичный ключ |
| id\_sector | int | - | + |
| id\_city | int | - | + |
| id\_street | int | - | + |
| number\_build | int | + | - |
| id\_ATM | int | - | + |

Таблица 4. Сущность “Street”

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Разрешить значения Null | Первичный ключ |
| id\_street | int | - | + |
| name\_street | nvarchar(50) | + | - |

Таблица 5. Сущность “Group\_Sector”

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Разрешить значения Null | Первичный ключ |
| id\_group | int | - | + |
| id\_sector | int | - | + |

Таблица 6. Сущность “Group\_”

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Разрешить значения Null | Первичный ключ |
| id\_group | int | - | + |
| id\_city | int | + | - |

Таблица 7. Сущность “Group\_Person”

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Разрешить значения Null | Первичный ключ |
| id\_group | int | - | + |
| id\_person | int | - | + |

Таблица 8. Сущность “Person”

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Разрешить значения Null | Первичный ключ |
| id\_person | int | - | + |
| second\_name | nvarchar(50) | + | - |
| name\_ | nvarchar(50) | + | - |
| patronymic | nvarchar(50) | + | - |
| photo | image | + | - |

Таблица 9. Сущность “City”

|  |  |  |  |
| --- | --- | --- | --- |
| Имя столбца | Тип данных | Разрешить значения Null | Первичный ключ |
| id\_city | int | - | + |
| name\_city | nvarchar(50) | + | - |

Ниже представлена Информационная модель базы данных (Рисунок 2)



Рисунок 2. Схема данных в базе данных

Составим проекты таблиц, то есть определим и опишем структуру базы данных, которая будет состоять из таблиц (Рисунок 3 - 8).

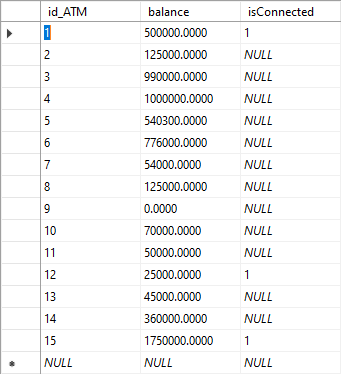


Рисунок 3. Таблица “ATM”

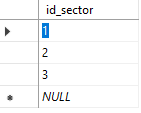


Рисунок 4. Таблица “Sector”

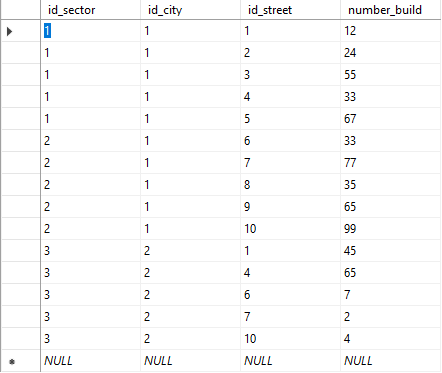


Рисунок 5. Таблица “Sector\_GPS”

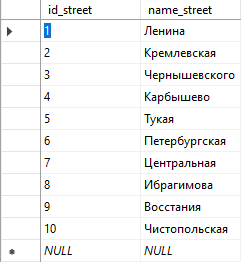


Рисунок 6. Таблица “Street”

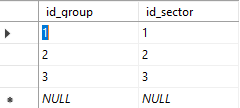


Рисунок 7. Таблица “Group\_Sector”

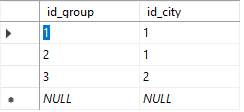


Рисунок 8. Таблица “Group\_”

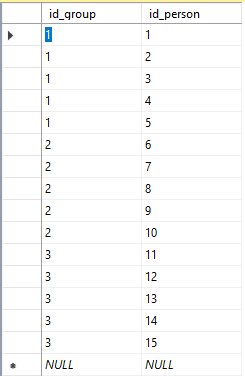


Рисунок 9. Таблица “Group\_Person”

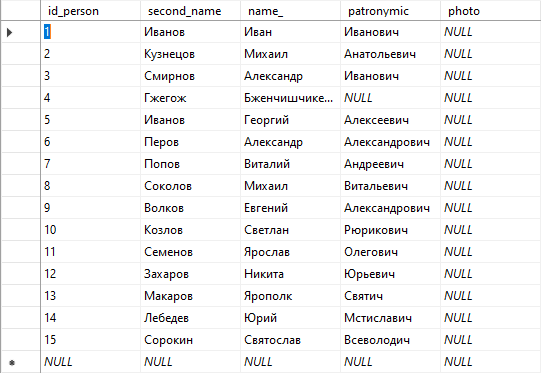


Рисунок 10. Таблица “Person”

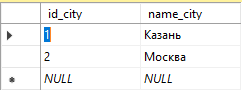


Рисунок 11. Таблица “City”

4.2 Компоненты, использованные в программе

Элементы управления представляют собой визуальные классы, которые получают введенные пользователем данные и могут инициировать различные события. При создании данной курсовой работы были использованы следующие компоненты:

* TextBlock
* Button
* Border
* ListBox
* ComboBox
* TextBox
* Frame
* Image

Элемент TextBlock - обеспечивает гибкую поддержку текста для сценариев пользовательского интерфейса, требующих не более одного абзаца текста. Он поддерживает ряд свойств, которые обеспечивают точный контроль над отображением текста, например, FontFamily, FontSize, FontWeight, TextEffects и TextWrapping. Содержимое текста может быть добавлено с помощью свойства Text. При использовании в XAML содержимое между открывающим и закрывающим тегами неявно добавляется в качестве текста элемента.

Элемент Button – элемент управления «Кнопка Windows», который будет реагировать на событие “Click”.

Элемент Border – элемент-декоратор, который можно использовать для рисования границы, фона или всего сразу вокруг другого элемента.

Элемент ListBox – элемент управления, позволяющий отображать данные в виде плитки.

Элемент ComboBox – образует выпадающий список и совмещает функциональность компонентов ListBox и TextBox.

Элемент TextBox – элемент управления пользовательской формы, предназначенный для ввода, редактирования и вывода информации в текстовом формате.

Элемент Image – предназначен для показа изображений.

Элемент Frame – предназначен для создания контейнера, содержащий встроенную навигацию.

# 5. Руководство пользователя

Для запуска и проведения работ нам потребуется компьютер или ноутбук с такими системными требованиями, как:

- ОС: Windows 7/8/10;

- Памяти диска HDD, либо SSD: 1ГБ;

- Процессор: четыре физических ядра с частотой от 2.7 ГГц;

- ОЗУ: не менее 4ГБ;

- Клавиатура;

- Компьютерная мышь;

- Монитор.

Запускаем программы: Microsoft SQL Server Management Studio 2018 и Visual Studio 2019

# 5.1 Руководство пользователя

Страница «PageATM» (Рисунок 9) позволяет видеть и отслеживать подключенные банкоматы. После нажатия кнопки «Подключить», совершается переход на другую страницу «PageConnectionATM» (рисунок 10).

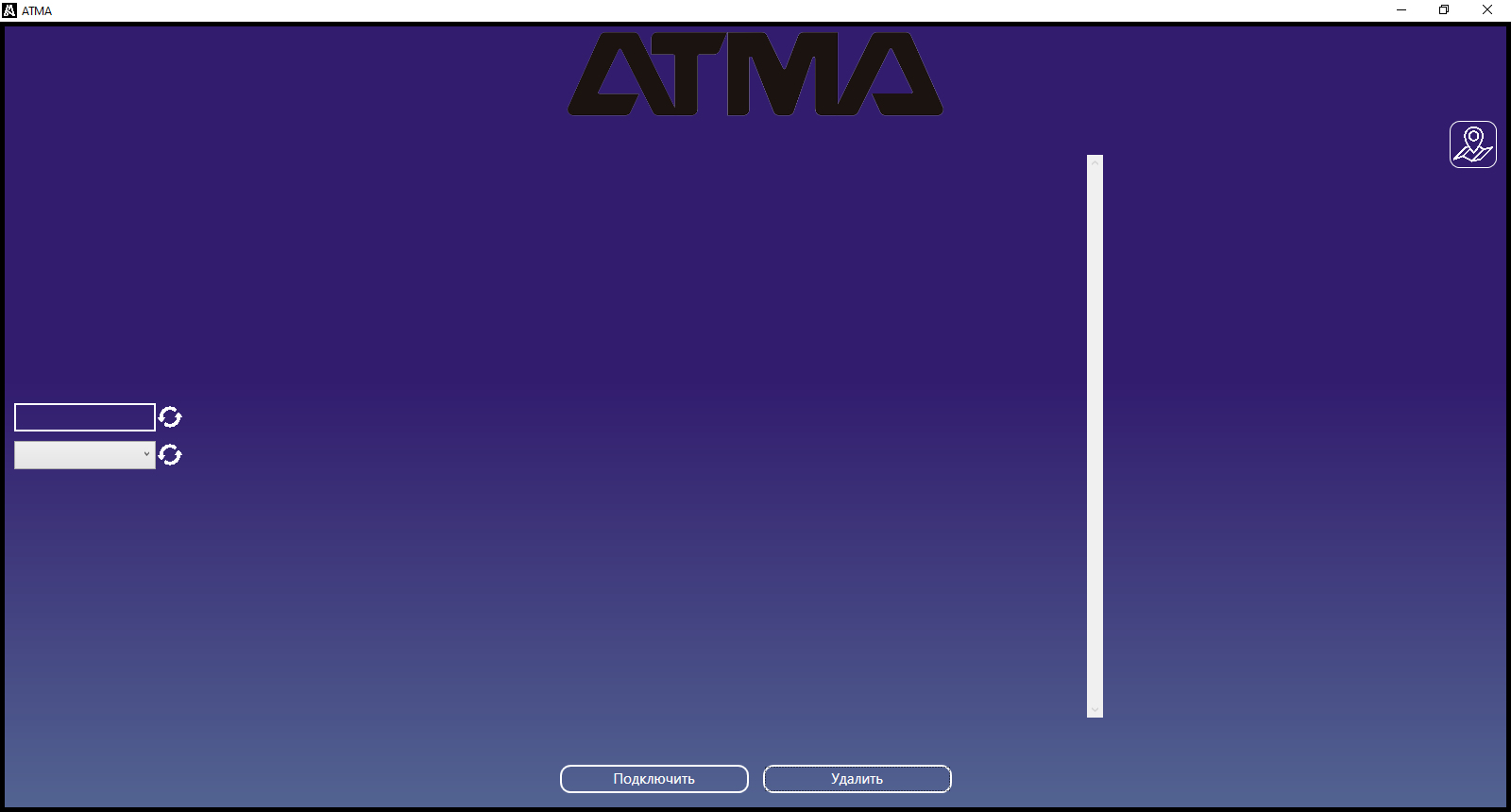


Рисунок 9. Страница «PageATM»

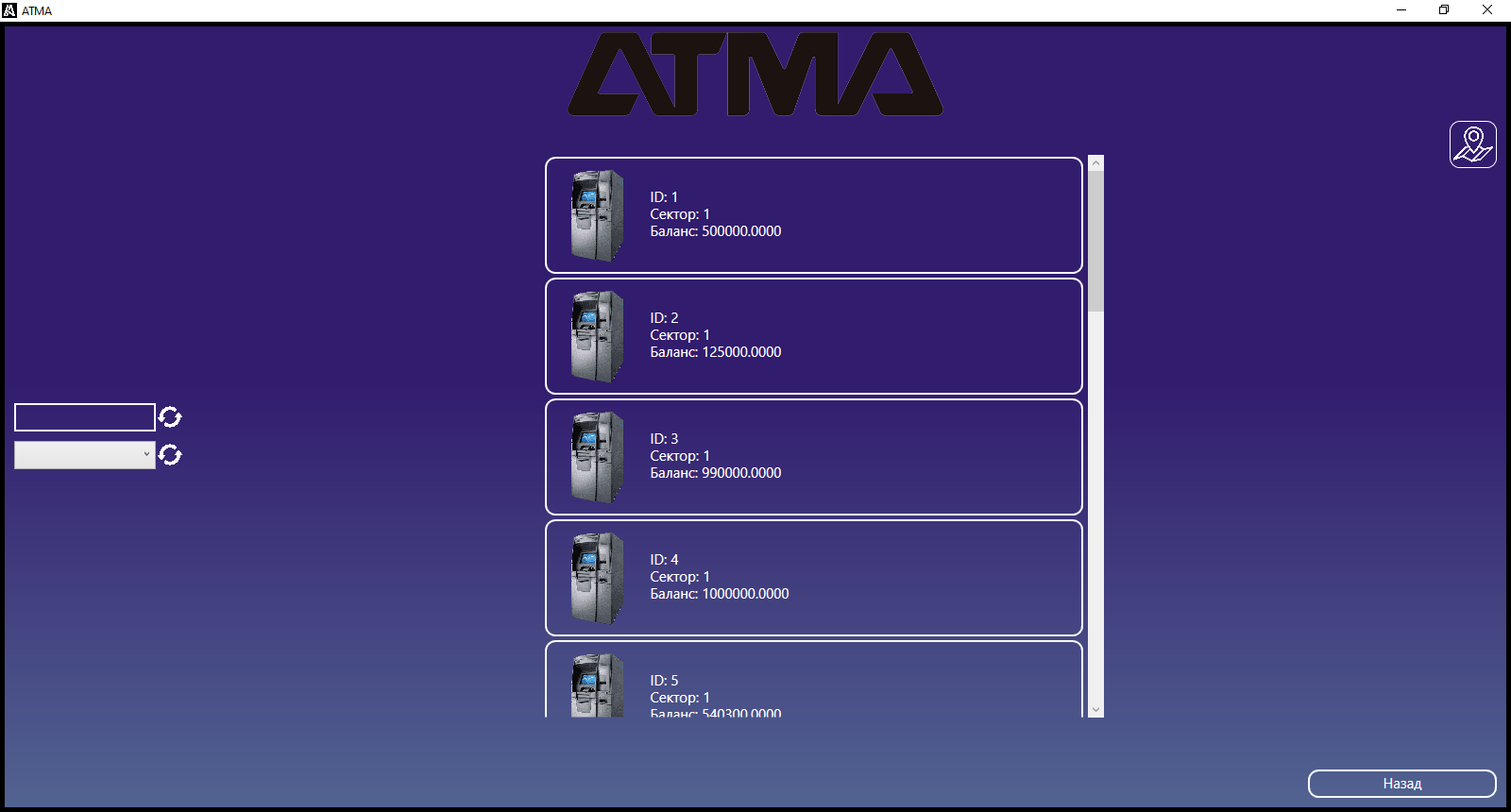


Рисунок 10. Страница «PageConnectionATM»

На рисунке 10 представлена страница базы банкоматов, где представлена информация о каждом. При двойном нажатии ЛКМ происходит добавление банкомата на главную страницу для его отслеживания(рисунок 11).

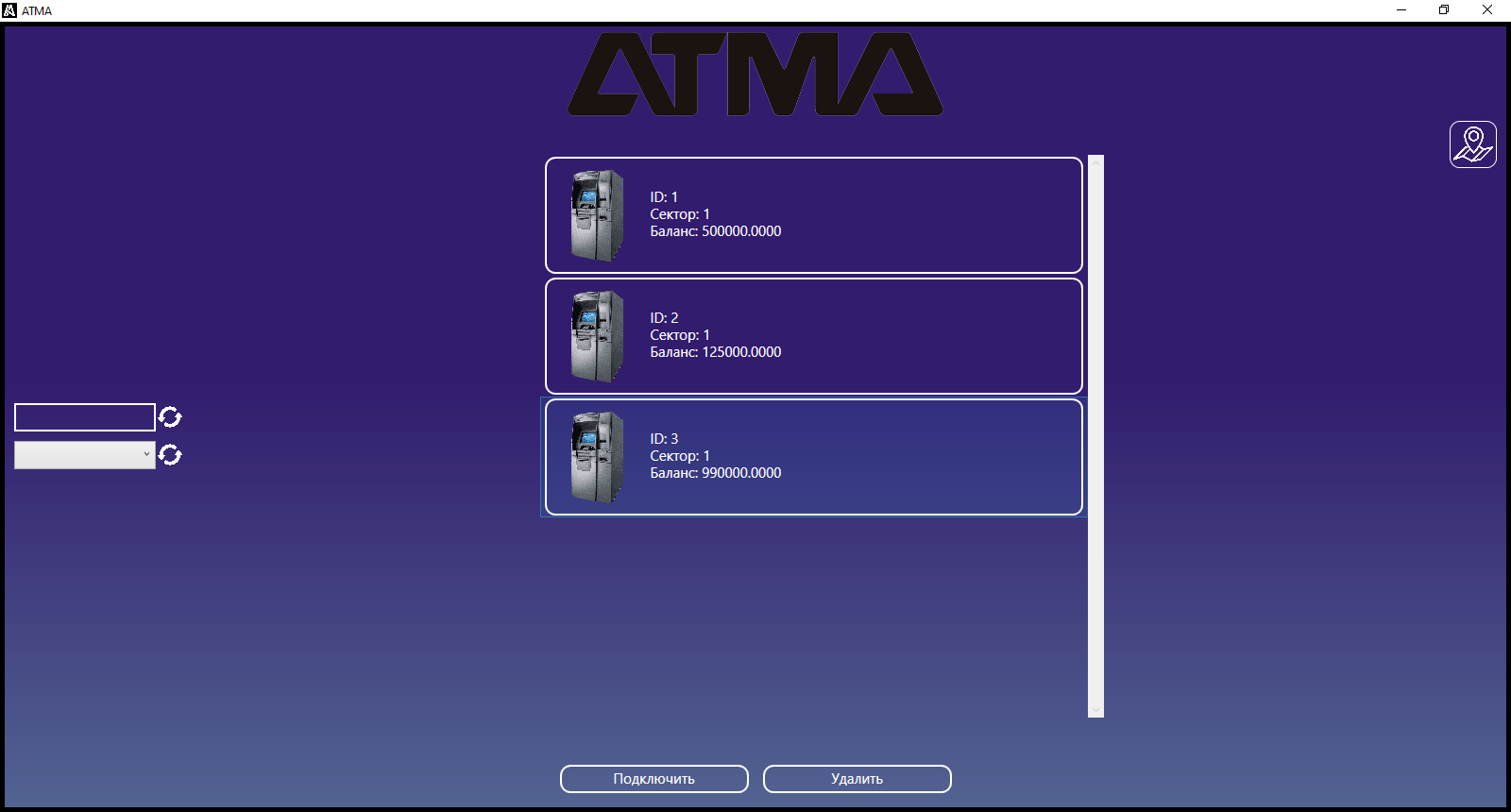


Рисунок 11. После добавления банкомата в список отслеживания

На этой странице так же представлен функционал для удобного пользования. Например, поиск, сортировка (рисунки 12-14)

Кнопка поиска (рисунок 12) с помощью нее желающий сможет найти банкомат по ID.



Рисунок 12. Поиск

Сортировка (рисунок 13) с помощью нее посетитель сможет сортировать банкоматы по секторам



Рисунок 14. Сортировка

Следующая страница (рисунок 16) представляет из себя полную информацию и возможностью просмотреть группу.

Так же кнопка назад, с помощью которой можно вернуться на страницу, описанную выше.

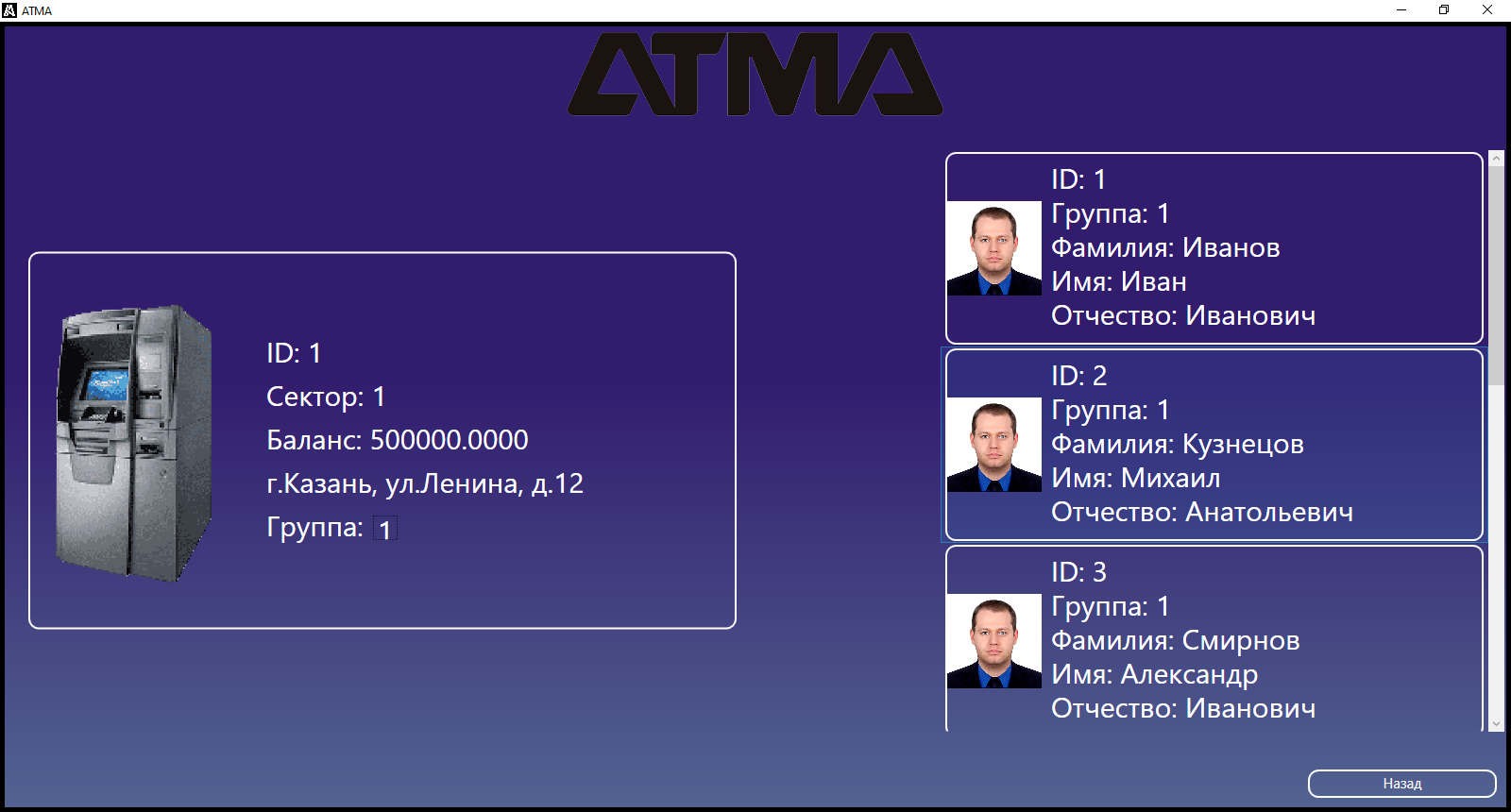


Рисунок 16. Страница выведении информации.

Следующая страница (рисунок 17) открывается после нажатия на карту, в которой возможно просмотреть местность возле банкомата. Так же есть кнопка назад, чтобы вернуться обратно к предыдущей странице.



Рисунок 17. Страница редактирования

# Заключение

За последние двадцать лет значительно возрос объём и оборот информации во всех сферах жизнедеятельности человека: экономической, финансовой, политической, духовной. И процесс накопления, обработки и использования знаний постоянно ускоряется. Учёные утверждают, что каждые десять лет количество информации увеличивается вдвое. В связи с этим возникает необходимость использования автоматических средств, позволяющих эффективно хранить, обрабатывать и распределять накопленные данные.

Исходя из современных требований, предъявляемых к качеству работы любого предприятия, нельзя не отметить, что эффективная работа его всецело зависит от уровня оснащения компании информационными средствами на базе компьютерных систем автоматизированного учета.

Компьютерный учет имеет свои особенности и радикально отличается от обычного. Компьютер не только облегчает учет, сокращая время, требующееся на оформление документов и обобщение накопленных данных для анализа хода торговой деятельности, необходимого для управления ею. Отчеты о положении в отрасли, получаемые с помощью компьютера, можно получить и без него, но на это требуется большее количество времени, трудовых и материальных затрат. Таким образом при применении компьютера “количество переходит в качество”: увеличение скорости расчетов делает возможным качественное улучшение работы предприятия.

Основное преимущество автоматизации — это сокращение избыточности хранимых данных, следовательно, экономия объема используемой памяти, уменьшение затрат на многократные операции обновления избыточных копий и устранение возможности возникновения противоречий из-за хранения в разных местах сведений об одном и том же объекте, увеличение степени достоверности информации и увеличение скорости обработки информации. Также значительно сокращает время автоматический поиск информации.

Информационная система позволит избавить сотрудника от рутинной повседневной работы по выписке расходных накладных. Так как раньше документы выписывались вручную, в которых указывались повторяющиеся реквизиты, а также большой перечень номенклатуры – это занимало много времени. Автоматизация позволит значительно сократить время.

За счет сокращения времени на выполнение долгих рутинных работ, можно повысить трудоемкость сотрудника, который может теперь выполнять не только свою работу, но и взять на себя ряд других обязанностей.

Создание собственной автоматизированной системы позволит учесть все особенности, разрабатывается только то, что нужно, и как нужно.

Целью курсовой работы являлась создание автоматизированного рабочего место для сотрудника по отслеживанию банкоматов.

Цель полностью выполнена, создана информационная система по отслеживанию(учет) банкоматов «ATMA».

# Библиографический список

1. Федеральный закон от 20.02.1995 №24-ФЗ «Об информации, информатизации и защите информации»// СЗ РФ 1995. №8 Ст.609.
2. Федеральный закон от 10.01.2002 №1-ФЗ «Об электронной цифровой подписи»// СЗ РФ 2002. №2 Ст.127.
3. Федеральный закон от 27.12.2002 №184-ФЗ «О техническом регулировании»// СЗ РФ 2002. №52 Ст.5140.
4. Баканов М. И. Шеремет А.Д. Теории экономического анализа - М: Финансы и статистика. 2020
5. Барнгольц С. Б. Экономической анализ хозяйственной деятельности предприятий и объединений – М.: Финансы и статистика. 2020
6. Баронов В.В. Автоматизация управления предприятия. –М.: Инфо-М, 2020
7. Берновский Ю.Н. Стандарты и качество. – М.: Из-во. Стандартов, 2022
8. Богатко А. Н. Основы экономического анализа хозяйствующего субъекта. – М.: Финансы и статистка. 2021
9. Буч Г. Объектно-ориентированное проектирование с примерами применения – М.: Радио и связь. 2021
10. Быкова Т.А. Документирование организаций и деятельности органов управления в акционерных обществах//Секретарское дело. № 2.
11. Быкова Т.А. Документирование процесса государственной регистрации акционерных обществ//Секретарское дело. 2021. № 8.
12. Вейцмен К. Распределенные системы мини- и микороЭВМ. – М.: Финансы и статистика. 2021
13. Вендров А. М. CASE-технологии. Современные методы и средства проектирования информационных систем. – М.: Финансы и статистика. 2018
14. Вирт Н. Алгоритмы и структура данных. – М.: Мир. 2017
15. WPF: Windows Presentation Foundation в .Net 4.5. Мэтью Макдональда 2020