|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  | Министерство науки и высшего образования Российской Федерации  Федеральное государственное бюджетное образовательное учреждение  высшего образования  Российский экономический университет им. Г.В. Плеханова  Московский приборостроительный техникум  Специальность: 09.02.05 Прикладная информатика (по отраслям)  Профессиональный модуль: ПМ 02 Разработка, внедрение и адаптация программного  обеспечения отраслевой направленности  Междисциплинарный курс: МДК 2.01 Разработка, внедрение и адаптация программного  обеспечения отраслевой направленности  МПТ.И-3-17-КП.РВиАПООН 16 20 - ЛУ  КУРСОВОЙ ПРОЕКТ  Тема: Разработка, внедрение и адаптация информационной системы: «Магазин оружия».  Пояснительная записка  Листов: 100   |  |  | | --- | --- | | Руководитель: | Выполнил: | | И.М. Щаников\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Студент группы И-3-17 | | «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г. | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Д.И. Темченко | | Оценка: \_\_\_\_\_\_\_(\_\_\_\_\_\_\_\_\_\_\_\_) | «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2020 г. |   2020 г. |
| Подп. N дата |  |
| Инв. N дубл |  |
| Взаим. Инв. N |  |
| Подп. N дата |  |
| Инв. N подл. |  |

Оглавление

[**ВВЕДЕНИЕ** 13](#_Toc43717111)

[**1.** **ОБЩАЯ ЧАСТЬ** 14](#_Toc43717112)

[**1.1 Цели разработки:** 14](#_Toc43717113)

[**1.2.** **Проблемная зона** 15](#_Toc43717114)

[**1.2.1 Описание предметной области:** 15](#_Toc43717115)

[**1.2.2 Описание подсистем:** 17](#_Toc43717116)

[**1.2.3 Описание информационных объектов** 18](#_Toc43717117)

[**1.3.** **Функциональное назначение разработки** 19](#_Toc43717118)

[**1.3.1.** **Схема бизнес процессов** 19](#_Toc43717119)

[**1.3.2.** **Модель потоков данных** 23](#_Toc43717120)

[**1.3.3 Событийная модель действий** 26](#_Toc43717121)

[**1.3.4 Дизайн проекта** 30](#_Toc43717122)

[**1.4. Эксплуатационное назначение разработки** 34](#_Toc43717123)

[**1.4.1. Роли** 34](#_Toc43717124)

[**1.4.2. Требования к параметрам технических средств** 36](#_Toc43717125)

[**2. СПЕЦИАЛЬНАЯ ЧАСТЬ** 38](#_Toc43717126)

[**2.1. Постановка задачи** 38](#_Toc43717127)

[**2.2. Проектирование** 39](#_Toc43717128)

[**2.2.1. Внутренняя спецификация** 39](#_Toc43717129)

[**2.2.2. Внешняя спецификация** 44](#_Toc43717130)

[**2.2.2.1. Функциональная схема (Рисунок 2.4)** 44](#_Toc43717131)

[**2.2.2.2. Алгоритмы** 44](#_Toc43717132)

[**2.2.2.3. Шифрование данных (Рисунок 2.11)** 48](#_Toc43717133)

[**2.2.2.4. Целостность данных** 49](#_Toc43717134)

[**2.2.2.5. Структурная схема (Рисунок 2.12)** 55](#_Toc43717135)

[**2.2.2.6. Схема пользовательского интерфейса (Рисунок 2.13)** 56](#_Toc43717136)

[**2.2.2.7. Тесты** 56](#_Toc43717137)

[**3. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ** 62](#_Toc43717138)

[**3.1. Средства разработки** 62](#_Toc43717139)

[**3.1.1. Средства анализа** 62](#_Toc43717140)

[**3.1.2. Средства проектирования** 62](#_Toc43717141)

[**3.1.3. Средства реализации** 62](#_Toc43717142)

[**3.1.4. Методы отладки** 65](#_Toc43717143)

[**3.1.5. Методы адаптации и конфигурации** 66](#_Toc43717144)

[**3.2. Оценка качества программного продукта** 75](#_Toc43717145)

[**3.2.1. Модель оценивания** 75](#_Toc43717146)

[**3.3. Итоги разработки** 95](#_Toc43717415)

[**ЗАКЛЮЧЕНИЕ** 97](#_Toc43717424)

[**СПИСОК ИСПОЛЬЗУЕМЫХ МАТЕРИАЛОВ** 98](#_Toc43717425)

# **ВВЕДЕНИЕ**

Информационная система – совокупность приложений, которые обеспечивают функционирование бизнес-процессов в организации. Информационная система «Магазин оружия» будет предназначена для упрощения заполнения, изменения и удаления временных и трудовых ресурсов предприятия. В подсистеме «Отдел продаж» – будет осуществлена возможность занесения зарегистрированных клиентов и проданных товаров в базу данных. В подсистеме «Отдел снабжения» – будет автоматизирована возможность хранения копий договоров с поставщиками и инвесторами. В подсистеме «Отдел кадров» – будет частично автоматизирован как набор, так и учет имеющегося персонала. Для реализации информационной системы «Магазин оружия» в рамках курсового проекта МДК 02.01 «Разработка, внедрение и адаптация программного обеспечения отраслевой направленности», будет проведен анализ предметной области, анализ протекающих бизнес-процессов по методологиям IDEF, DFD, EPC, будет разработан дизайн проект программного обеспечения, а затем и само программное обеспечение. После будут проведены действия по внедрению и адаптации программного обеспечения в организацию, будут произведены тесты ПО, будет подготовлена оценка качества продукта и документация.

1. **ОБЩАЯ ЧАСТЬ**

## **1.1 Цели разработки:**

1. Закрепление теоретических навыков в рамках МДК 02.01 Разработка, внедрение и адаптация программного обеспечения отраслевой направленности;
2. Закрепление практических умений в рамках МДК 02.01 Разработка, внедрение и адаптация программного обеспечения отраслевой направленности;
3. Закрепление и более глубокое освоение профессиональных компетенций в рамках профессионального модуля ПМ 02 Разработка, внедрение и адаптация программного обеспечения отраслевой направленности;
4. Получение опыта в работе с условной предметной областью «Магазин оружия»;
5. Проведение анализа предметной области «Магазин оружия» для выявления оптимального количества выполняемых функций в ИС;
6. Проведение и реализация проектной специфики предметной области «Магазин оружия»;
7. Автоматизация бизнес – процессов предприятия «Магазин оружия»;
8. Обеспечение «Отдела кадров»;
9. Упрощение «Работы отдела снабжения»;
10. Сокращение по времени «Работы отдела снабжения»;
11. Обеспечение документооборота отделов предприятия «Магазин оружия»;
12. Освоение методов отладки ручного тестирования, индукции, дедукции, обратного прослеживания;
13. Закрепление навыков выявлении ошибок реализуемого ПО ИС «Магазин оружия;
14. Реализация адаптивного программного продукта ИС «Магазин оружия»;
15. Реализация конфигурируемого программного продукта ИС «Магазин оружия»;
16. Проведение оценки качества информационной системы «Магазин оружия»;
17. Закрепление опыта в разработке технической документации к информационной системе «Магазин оружия;
18. Получение опыта в разработке первого курсового проекта.
    1. **Проблемная зона**

**1.2.1 Описание предметной области:**

Магазин оружия представляет собой хранилище оружия. Основная

функции магазина-продавать оружие при наличии у покупателя лицензии на его приобретение. Экземпляр каждого орудия должен иметь индивидуальный номер. Каждый покупатель обязан предъявить лицензию на разрешения приобретение оружия и паспорт. Она должна выглядеть как бумажный бланк, состоять из 3-частей. В каждой части на лицевой стороне указан уникальный номер, личные данные и тип оружия доступный для приобретения, а на обороте графы для указания конкретного оружия, которое будет куплено, а также её цвет в зависимости от типов оружия. Размер А4 в случае с газовым оружием А6. Если лицензия будет не отвечать каким-либо из перечисленных требований, отдел безопасности будет следовать стандартным инструкциям при фальсификации документов. После её предъявления, в БД заносится электронная версия лицензии, которая имеет свой номер, дату и информацию о покупателе. При невозможности занесения электронной версии покупателю будет отказано в приобретении или назначен срок следующего посещения магазина. Только после этого покупатель имеет возможность выбрать себе оружие из ассортимента товаров в соответствии с типом оружия, указанным в лицензии. После выбора товара клиент может протестировать оружие в тире, находящимся на втором этаже нашего магазина или же прямо у кассы произвести частичную разборку, проверить механизмы и т.д. Если качество товара устраивает клиента происходит процесс оформления товара: покупатель проходит на кассу, где оплачивает покупку и заполняются графы лицензии об купленном оружии. Покупатель расписывается в специальной графе, и лицензия разрезается на 3 части. Две части лицензии магазин оставляет у себя, а часть с подписью остаётся у покупателя. Возвращается паспорт, оружие упаковывается и отдаётся клиенту и передаётся протокол контрольного отстрела, если оружие было нарезным. Патроны покупается отдельно по лицензии на хранение аналогичным путём. Купленное оружие не подлежит возврату.

Каждое оружие характеризуется следующими параметрами: уникальный номер, обладает своим названием, является определённым типом (например, гладкоствольная винтовка), серия, год издания, страной поставки, ценой и количеством экземпляров. На каждого покупателя в книгу продаж заносятся следующие сведения: ФИО, домашний адрес, телефон, дата рождения и номер паспорта, номер лицензии и купленное им оружие.

Каждый сотрудник, принятый на работу должен иметь среднее образование по специальности «Товаровед». При собеседовании на работу соискатель обязан предъявить паспорт, трудовую книжку, диплом об образовании и своё резюме. После собеседования, кандидатура соискателя рассматривается менеджером по отделу кадров и принимается решение о приёме или отказе на принятие соискателя на работу. Если сотрудник принят на работу, ему выдаётся порядковый номер с электронной карточкой для фиксирования его рабочей деятельности (выход на работу, перерывы и уход). Далее составляется менеджером по отделу кадров обновленное рабочее расписание с включением смен нового работника. Если работник захотел уволиться с работы, то он обязан написать письмо директору магазина или прийти лично для обговорения причины увольнения. В письме обязательно указывается причина увольнения. Если сотрудника увольняет начальство (например, воровство), то в письме указывается причина увольнения, назначенная начальством. При увольнении работника его смены вычёркиваются из расписания.

Каждая поставка оружия, чётко производится по заранее составленному договору и осуществляется в его рамках. После прибытия поставки на территорию магазина поставки оружия работники магазина по расписанию разгружают товар на склад, где оружие будет хранится до его продажи или списания из-за непригодности. После разгрузки товара производится занесение электронных версий договоров сторон в БД и только после этого происходит складской учёт. В электронную версию договора записывается его номер, дата подписания, тип и название. При невозможности создания электронных версий товар хранится на складе без возможности его продажи и проведения учёта. Со стороны компании-поставщика подписывается договор о поставке, а со стороны магазина подписывается договор о принятии товара.

**1.2.2 Описание подсистем:**

Отдел кадров – данная подсистема предназначена для управления персоналом магазина с помощью менеджера по кадрам, который составляет трудовой план, а именно: составление расписания смен для работников, исходя из их возможностей, составление графиков отпусков и перерывов. Также менеджер проводит собеседования для приёма на работу новых сотрудников и оформляет их увольнение.

Отдел продаж - роль данной подсистемы заключается в формировании истории продаж. В неё заносятся все сведения о реализованных товарах (название оружие, cсерийный номер, тип оружия, модель, дата поставки, время продажи, цена оружия, кассир, осуществивший продажу, лицензиях покупателей по которым была осуществлена продажа и личные данные клиентов.

Отдел снабжения - подсистема взаимосвязана с отделами кадров и продаж и выполняет следующие задачи такие как: 1) Поставка оружия в магазин и постановка его на складской учёт; 2) Обеспечение должной транспортировки оружия-транспортировка осуществляется со склада компании-поставщика на склад магазин. Товар перевозится в грузовой машине с символикой компании поставщика. Оружие разных типов транспортируется по-разному: стрелковое оружие перевозится в чехлах-контейнерах, патроны в коробках, заряженных магазинах или патронташах, а газовое или взрывоопасное оружие такое как, гранаты перевозятся в отдельных контейнерах с мягкой набивкой для амортизации при транспортировке. После приезда машины на территорию магазина производится разгрузка товара на склад магазина с последующим подписанием соответствующих бумаг. После всех вышеперечисленных действий, разгруженная машина отправляется обратно в компанию поставщика.; 3) Заключение договоров с поставщиками-договор подписывается на долгосрочные перспективы сотрудничества со следующими обязательными параметрами: 1) Предмет договора в нашем случае оружие, оружейные модификации и патроны; 2) Цена товаров; 3) Наличная система оплаты с последовательной оплатой заказа; 4) Место доставки и ответственное лицо, которое будет принимать партию. В данном место поставки будет сам магазин, а ответственным за принятие будет старший на смене менеджер. Он будет производить осмотр товара и немедленно сообщит поставщику, если будут выявлены какие-либо несоответствия. Далее мы осуществим отказ от принятия и организуем соответствующее хранение до того момента, когда поставщик заберёт товар и оплатит расходы за хранение.5) Тара и упаковка. Её вид был указан выше. 6) Срок действия договора поставки товара. Условия поставки. Также договор содержит реквизиты сторон такие как: 1) ФИО или наименование организации, место жительства или юридический адрес поставщика и покупателя; 2) Дата и место заключения договора; 3) Личные подписи (печати) поставщика и покупателя.

### **1.2.3 Описание информационных объектов**

Вакансия на работу – заявка, в которой отправляется перечень документов для принятия на работу.

Товар – оружие, патроны и оружейные модификации, которые выдаются клиентам при оплате и наличии у них лицензии.

Персонал- Сотрудники, которые предоставляют услуги магазина оружия клиентам.

Покупатель – совершеннолетний человек, пришедший в магазин купить товар по лицензии.

Потенциальные сотрудники по вакансии- люди, ищущие работу и готовые работать штатными работниками в магазине с возможностью повышения.

Проданное оружие – реализованный товар, занесённый в книгу продаж.

Новые сотрудники – люди, устроившиеся на работу в магазин по вакансии.

Оформленная лицензия– после продажи оружия, лицензия оформляется и закрепляется подписями со стороны продавца и покупателя печатями и росписями.

Новая запись в книги продаж– запись, занесённая после реализации товара и хранящая сведения о покупке.

Администрация–руководство магазина, отвечающее за организацию и деятельность магазина.

Кассовый аппарат–устройство, с помощью которого производится продажа товара.

Продавец-кассир– человек, с помощью которого производится продажа товара.

Правоохранительные органы–регламентирующие органы проверки, контролирующие деятельность магазина для предотвращения ситуаций нарушения закона.

Книга продаж– документ, хранящий информацию о реализованных товарах.

Расписание – расписание работы для сотрудников

Лицензия на приобретение оружия – документ, выданный государством и разрешающий клиенту приобрести оружие определённого типа.

Лицензия на хранение оружия – документ, выданный государством и влияющий на расширение круга товаров, которые покупатель имеет право купить в магазине (Например, патроны).

Договора с компаниями-инвесторами- договора о спонсировании магазина

Договора с поставщиками- договора о закупке и транспортировке товара на территорию магазина для дальнейшей реализации.

Трудовой договор-свод обязанностей работника, регламентированный конституцией.

Уголовный кодекс РФ – законы страны, направленные на предотвращение преступлений

Уволенные сотрудники - сотрудники предприятия, уволившиеся с работы.

* 1. **Функциональное назначение разработки**
     1. **Схема бизнес процессов**

- Иллюстрация родительской функции (предметная область) до реализации – Рисунок 1.1;

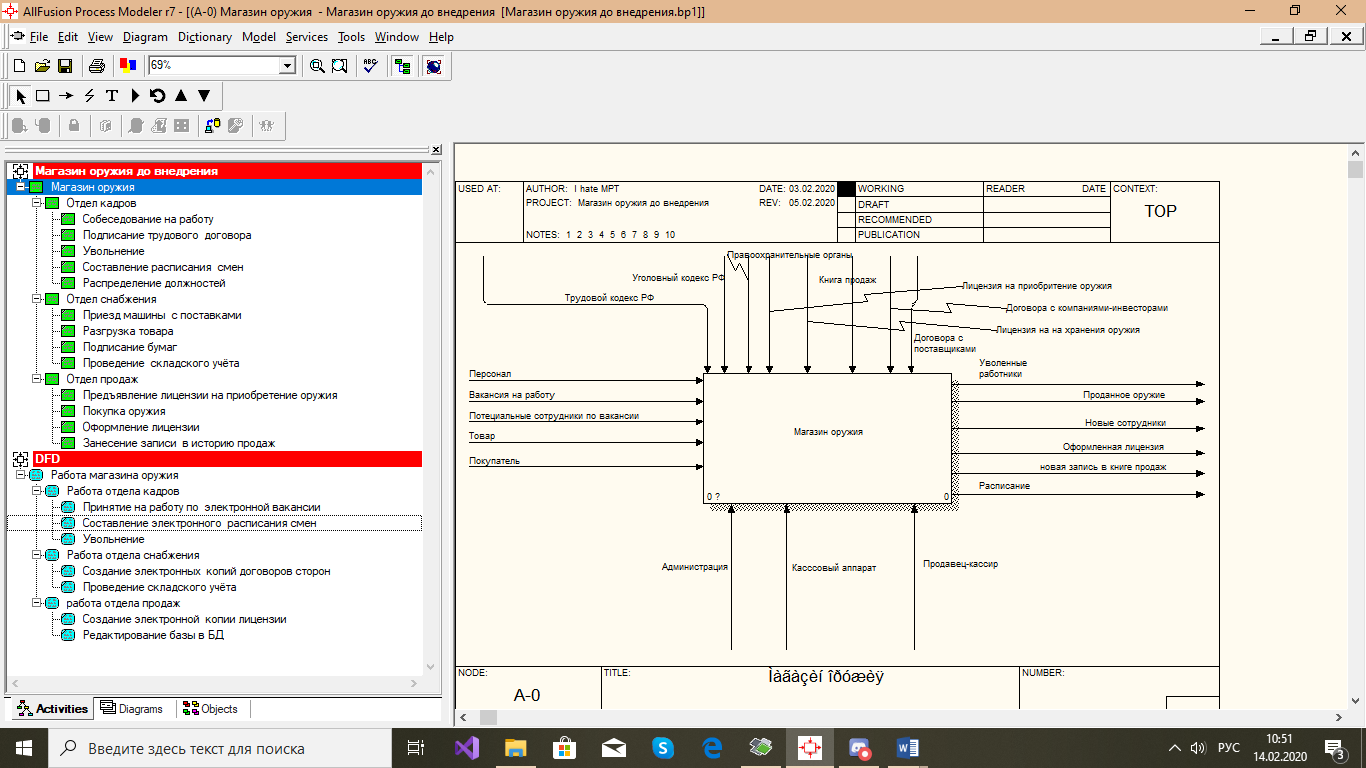


Рисунок 1.1 – Родительская функция IDEF0 до реализации

- Таблица с описанием всех объектов модели на первом уровне модели IDEF0 до реализации – таблица 1.1;

Таблица 1.1 Описание объектов предметной области модели IDEF0 до реализации

|  |  |  |
| --- | --- | --- |
| Элемент нотации | Имя объекта | Краткое описание объекта |
| Стрела входа | Персонал | Сотрудники, которые предоставляют услуги магазина оружия клиентам. |
| Вакансия на работу | Заявка, в которой отправляется перечень документов для принятия на работу. |
| Потенциальные сотрудники по вакансии | Люди, ищущие работу и готовые работать штатными работниками в магазине с возможностью повышения. |
| Товар | Оружие, патроны и оружейные модификации, которые выдаются клиентам при оплате и наличии у них лицензии. |
| Покупатель | Совершеннолетний человек, пришедший в магазин купить товар по лицензии. |
| Стрела выхода | Новые сотрудники | Люди, устроившиеся на работу в магазин по вакансии. |
| Проданное оружие | Реализованный товар, занесённый в книгу продаж. |
| Уволенные работники | Сотрудники предприятия, уволившиеся с работы |
| Оформленная лицензия | После продажи оружия, лицензия оформляется и закрепляется подписями со стороны продавца и покупателя печатями и росписями. |
| Новая запись в книге продаж | Запись, занесённая после реализации товара и хранящая сведения о покупке. |
| Расписание | Рабочий график для персонала |
| Стрела контроля | Трудовой кодекс РФ | Свод обязанностей работника, регламентированный конституцией |
| Уголовный кодекс РФ | Законы страны, направленные на предотвращение преступлений |
| Правоохранительные органы | Регламентирующие органы проверки, контролирующие деятельность магазина для предотвращения ситуаций нарушения закона. |
| Книга продаж | Документ, хранящий информацию о реализованных товарах. |
| Лицензия на приобретение оружия | Документ, выданный государством и разрешающий клиенту приобрести оружие определённого типа. |
| Лицензия на хранение оружия | Документ, выданный государством и влияющий на расширение круга товаров, которые покупатель имеет право купить в магазине (Например, патроны). |
| Договора с поставщиками | Договора о закупке и транспортировке товара на территорию магазина для дальнейшей реализации. |
| Договора с компаниями- инвесторами | Договора о спонсировании магазина |
| Стрела механизма | Кассовый аппарат | Устройство, с помощью которого производится продажа товара. |
| Администрация | Руководство магазина, отвечающее за организацию и деятельность магазина. |
| Продавец-кассир | Человек, с помощью которого производится продажа товара. |

- Иллюстрация декомпозиции родительской функции на три функции (подсистемы) до реализации – Рисунок 1.2.;

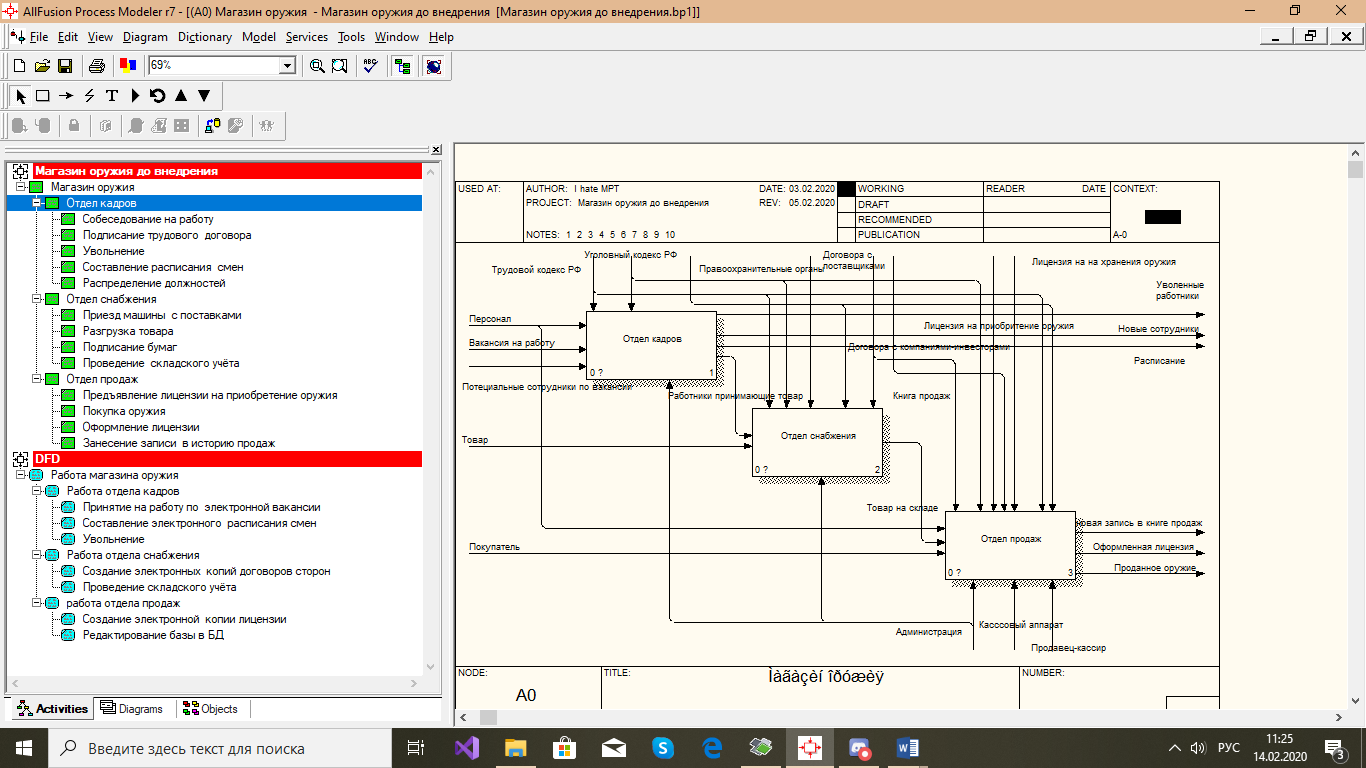


Рисунок 1.2 - декомпозиции родительской функции на 3 функции до реализации

- Таблица с описанием всех объектов модели на втором уровне модели IDEF0 до реализации – таблица 1.2;

Таблица 1.2. Описание объектов предметной области модели IDEF0 до реализации

|  |  |  |
| --- | --- | --- |
| Элемент нотации | Имя объекта | Краткое описание объекта |
| Стрела выхода | Работники, принимающие товар | Сотрудники предприятия, осуществляющие приём и разгрузку товара |
| Товар на складе | Разгруженный товар на складе |
| Стрела входа | Работники, принимающие товар | Сотрудники предприятия, осуществляющие приём и разгрузку товара перечне и стоимости услуг |
| Товар на складе | Разгруженный товар на складе |

Декомпозиция функций на уровни ниже описана в приложении А.

* + 1. **Модель потоков данных**

- Иллюстрация родительского процесса (предметная область), Рисунок 1.3;

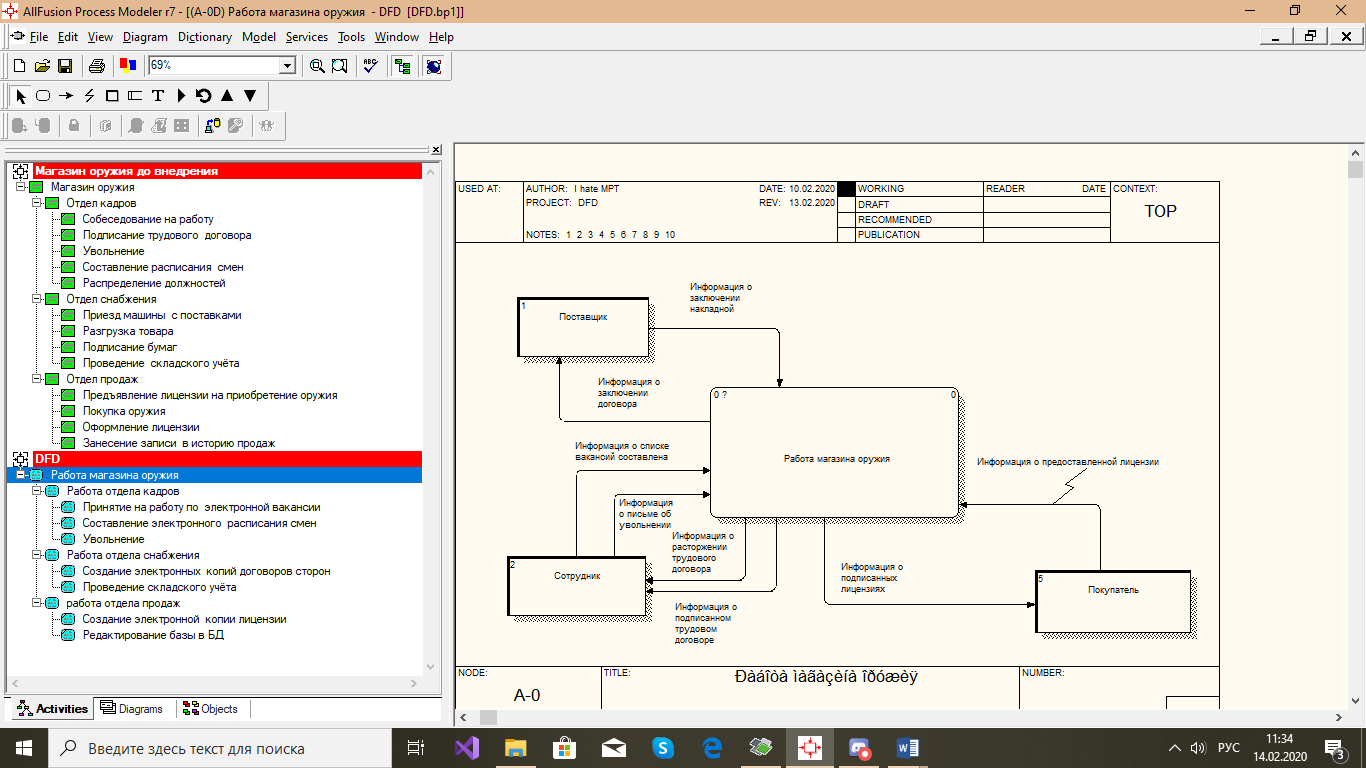


Рисунок 1.3 - Родительский процесс DFD

- Таблица с описанием всех объектов модели на первом уровне, таблица 1.3;

Таблица 1.3. Описание объектов предметной области модели DFD

|  |  |  |
| --- | --- | --- |
| Элемент нотации | Имя объекта | Краткое описание объекта |
| Поток данных | Информация о заключении накладной | Информация о поставляемых товарах, количество товаров, состояние поставки |
| Информация о заключении договора | Предмет заключения договора, дата подписание, юр. Название организаций-сторон, ФИО ответственных лиц |
| Информация о списке вакансий | Личное портфолио соискателя(сертификаты, диплом), список умений, ФИО, возраст |
| Информация о письме об увольнении | Причина увольнения сотрудника, дата написания, ФИО сотрудника. |
| Информация о расторжении трудового договора | Дата расторжения трудового договора, ФИО сотрудника, юр. Наименование организации |
| Информация о подписанном трудовом договоре | Дата заключения трудового договора, ФИО соискателя, юр. Наименование организации, условия и обязанности каждой из сторон |
| Информация о предоставленной лицензии | Номер лицензии, категория, ФИО лица получившего лицензию, дата получения. |
| Информация о подписанных лицензиях | Номер лицензии, категория, ФИО лица получившего лицензию, дата подписания. |
| Внешняя сущность | Сотрудник | Получает обработанную информацию от работы отдела кадров |
| Поставщик | Получает обработанные данные сферы поставки и снабжения |
| Покупатель | Получает обработанную информацию от работы отдела продаж |

- Иллюстрация детализации родительского процесса на три под процесса (подсистемы), Рисунок 1.4;

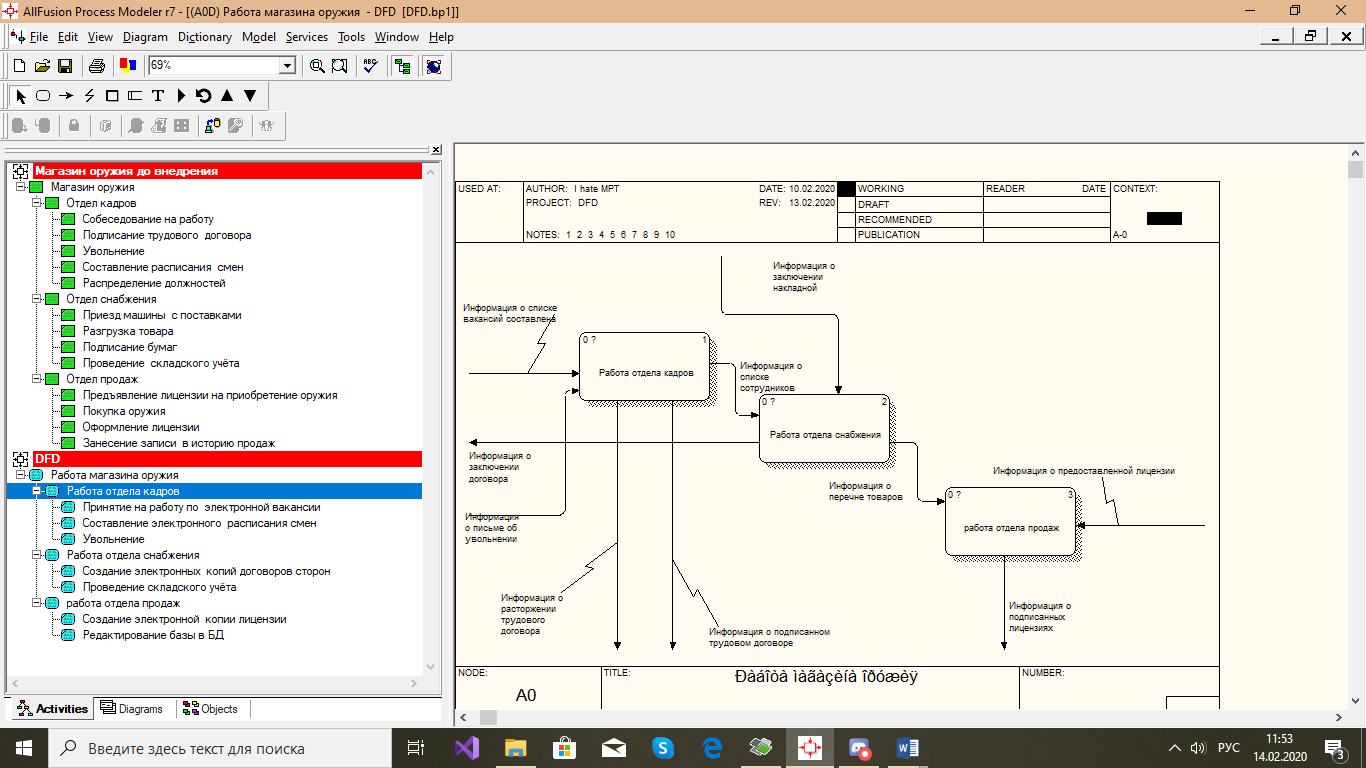


Рисунок 1.4 - декомпозиция 2 уровня родительского процесса

- Таблица с описанием всех объектов модели на втором уровне, таблица 1.4.

Таблица 1.4. Описание объектов подсистем предметной области модели DFD

|  |  |  |
| --- | --- | --- |
| Элемент нотации | Имя объекта | Краткое описание объекта |
| Поток данных | Информация о списке сотрудников | Количество сотрудников на предприятии, их должности и рабочее расписание |
| Информация о перечне товаров | Хранит в себе информацию о всех имеющихся товарах, готовых к реализации. |

Детализация процессов на уровни ниже описана в приложении Б.

### **1.3.3 Событийная модель действий**

Иллюстрация событийного алгоритма «Магазин оружия» с участием элементов «Process Interface», Рисунок 1.5;

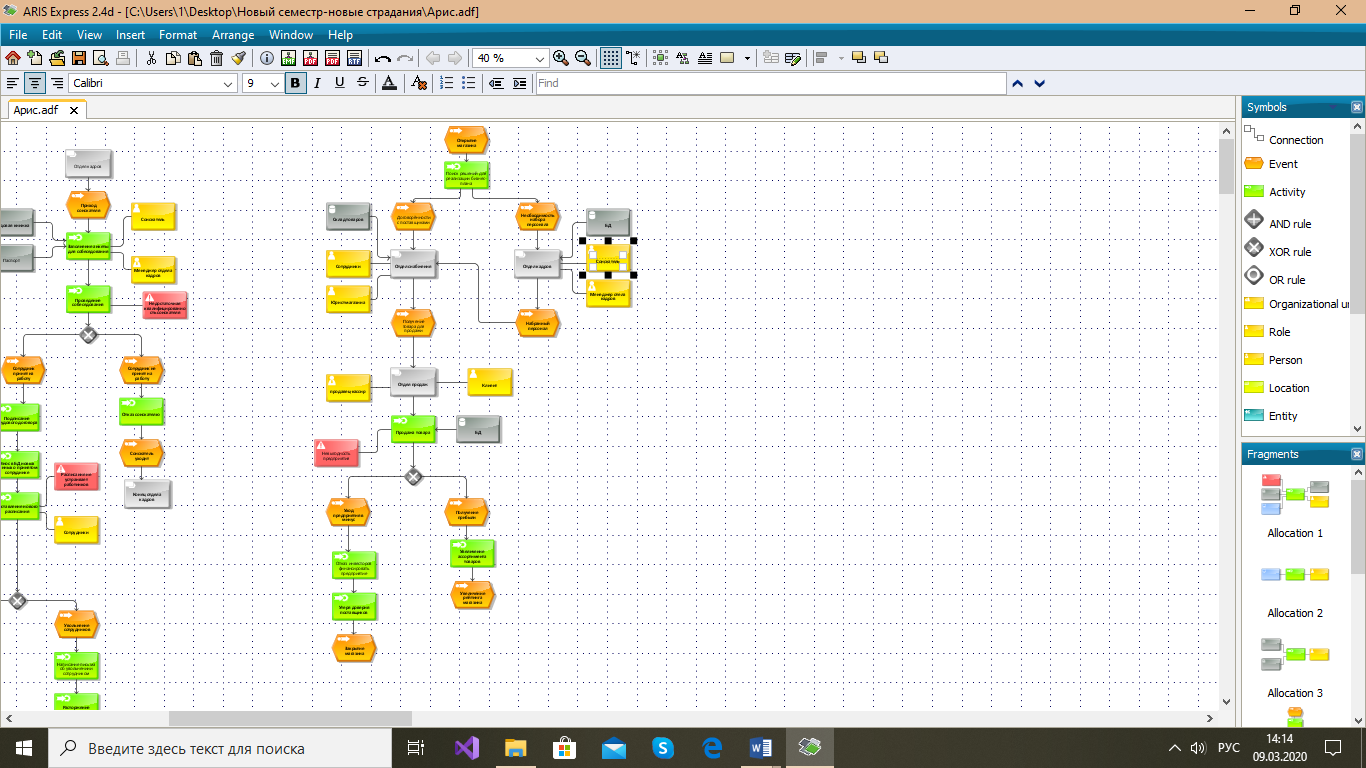


Рисунок 1.5 - событийный алгоритм предметной области «Магазин оружия»

Таблица с описание всех объектов событийной цепочки предметной области «Гостиница», таблица 1.5;

Таблица 1.5. Описание объектов EPC

|  |  |  |
| --- | --- | --- |
| Элемент нотации | Имя объекта | Краткое описание объекта |
| Событие | Открытие магазина | Здание магазина арендовано и готово к реализации бизнес плана |
| Договорённости с поставщиками | Заключение договоров о поставках |
| Необходимость набора персонала | Для дальнейшей работы магазина требуется набрать персонал |
| Набранный персонал | Персонал набран и готов к работе |
| Получение товара для продажи | Товар прибыл на склад предприятия и готов к реализации |
| Уход предприятия в минус | Работа предприятия уводит прибыль в минус |
| Получение прибыли | Предприятие в результате реализации товара получает прибыль |
| Увеличение рейтинга магазина | Реализованный товар нравится клиентам, что повышает рейтинг магазина |
| Закрытие магазина | Магазин обанкротился и закрыт из-за отсутствия денег на его дальнейшее развитие и невыгодности дальнейшего развития. |
| Действие | Поиск решений для реализации бизнес-плана | Процесс нахождения подходящих инвесторов, поставщиков, персонала и ассортимента реализуемого товара. |
| Продажа товара | Процесс продажи товара для получения прибыли |
| Отказ инвесторов финансировать предприятие | Процесс расторжения договорённостей с инвесторами |
| Утеря доверия поставщиков | Процесс расторжения договоров с поставщиками |
| Увеличение ассортимента товаров | Процесс подписания новых договоров с поставщиками на получение нового товара для расширения ассортимента. |
| Сотрудник отдела | Менеджер отдела кадров | Выполняет функцию по работе с персоналом. |
| Юрист магазина | Выполняет функцию подписания договорённостей и других юридических вопросов, касающихся дел предприятия. |
| Продавец-кассир | Выполняет функцию по работе с клиентом при продаже товара. |
| Сотрудники | Выполняют функции разгрузки/отгрузки товара, уборки помещений, чистки товара, охраны предприятия. |
| Человек, не относящийся к фирме | Соискатель | Предоставляет личные данные для трудоустройства на свободную вакансию |
| Клиент | Предоставляет личные данные для приема услуг, предоставляемых магазином оружия. |
| Фактор риска | Не востребованность предприятия | Товар, который реализует предприятие не востребован, либо здание находится в неудачной геолокации, либо поток клиентов слишком маленький и прибыль магазина ниже, чем расходы на содержание магазина. |
| Отдел организации | Отдел снабжения | Занимается поставками товара в магазин и подписанием договорённостей с поставщиками и инвесторами. |
| Отдел кадров | Занимается приемом на свободные вакансии соискателей, повышением квалификации сотрудников, их увольнением, составлением расписания, а так же следит за работой персонала. |
| Отдел продаж | Занимается продажей товара. |
| Источник данных | БД | Хранит данные о свободных вакансиях, сотрудниках, товарах, договорах и документах предприятия, трудовом графике. |
| Склад товаров | Хранит данные о товарах на складе и их статусе. |

Подробные алгоритмы подсистем описаны в приложении В.

### **1.3.4 Дизайн проекта**

1) Иконка программного продукта (Рисунок 1.6);



Рисунок 1.6 - Иконка

1. Стартовая заставка приложения (Рисунок 1.7);

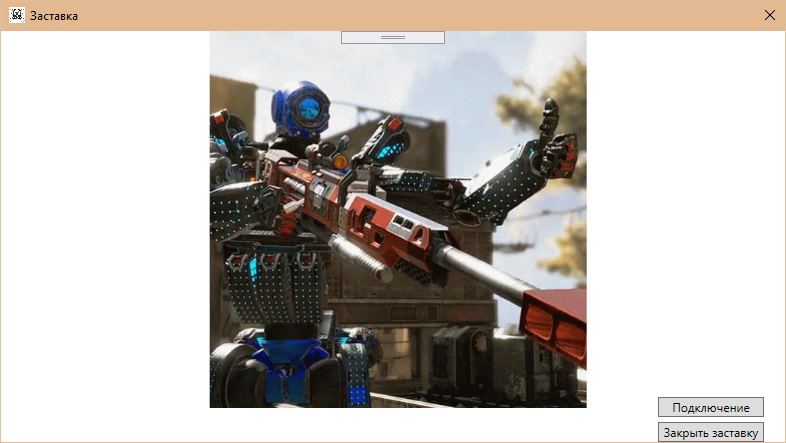


Рисунок 1.7 – Заставка

1. Окно аутентификации:

3.1) Авторизация (Рисунок 1.8);

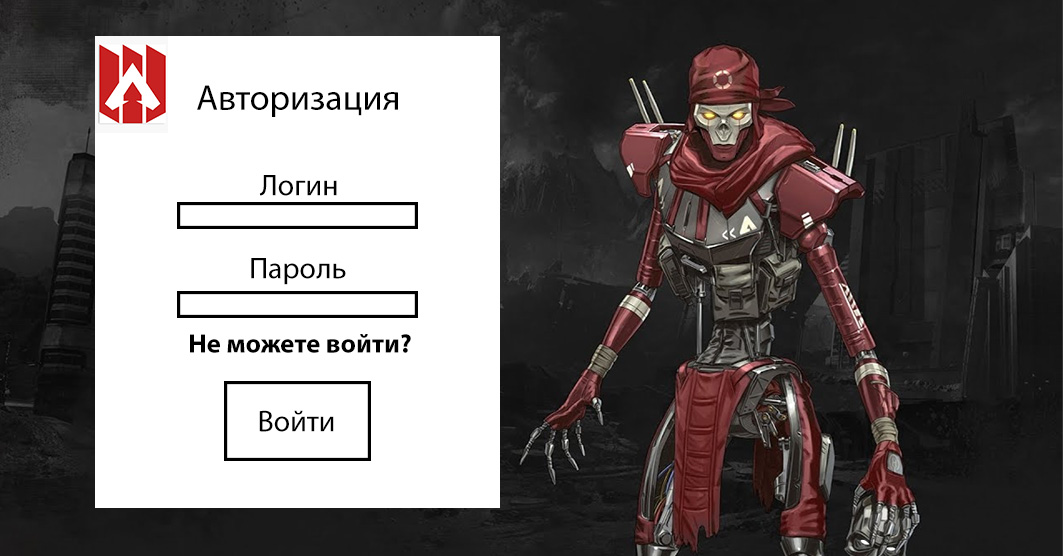


Рисунок 1.8 - Авторизация

3.2) Регистрация (Рисунок 1.9);

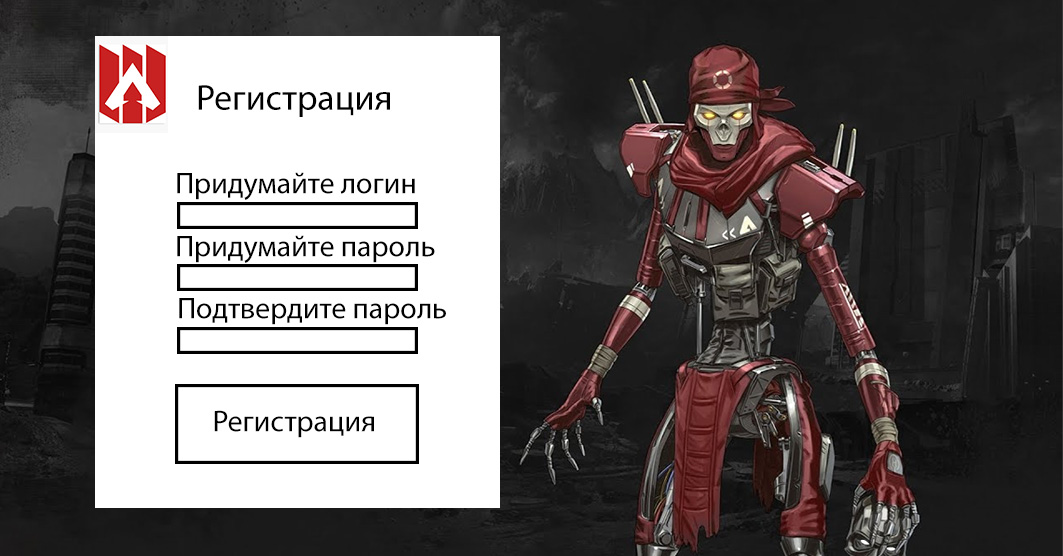


Рисунок 1.9 – Регистрация

1. Главное меню (Рисунок 1.10);



Рисунок 1.10 - главное меню

Сопоставление элементов дизайн проекта с событийной цепочкой действий методологии EPC (Рисунок 1.6)

Таблица 1.6. Сопоставление элементов методологии EPC с элементами дизайн проекта

|  |  |
| --- | --- |
| Элемент EPC | Элемент дизайн проекта |
| Ответственное лицо приезжает к поставщику для подписания договора->Рассмотрение условий договора->Условия устраивают обе стороны->Создание электронных копий договоров->Подписание оригинальных договоров->Составление поставщиком накладной->Выезд машины с поставкой | Окно для внесения новой электронной копии договора и накладной в БД |
| Машина приехала->проверка наличия товара по накладной->Товар в наличии полностью->Разгрузка машины->Товар разгружен->Проведение складского учёта->Занесение показаний складского учёта в БД | Окно внесения нового товара на складе в БД |

|  |  |
| --- | --- |
| Машина не приехала->Обращение к поставщику с просьбой разрешения проблемы->Поставщик не отвечает на просьбу->требование возврата денег->Поставщик отказывается вернуть денег->Расторжение договора->Обращение в суд->Возврат денег через судебный процесс | Окно для удаления договоров из БД в связи с их расторжением |
| Товар выбран->Оформление товара->Оплата товара->Деньги есть->Оформление лицензии->Упаковывание товара->Выдача покупателю оформленной лицензии и паспорта->Выдача покупателю оплаченного товара->Клиент уходит | Окно занесения товара со склада в книгу продаж и оформленных лицензий в БД |
| Приход соискателя->Заполнение анкеты для собеседования->Проведение собеседования->Сотрудник подходит->Подписание трудового договора->Внос в БД новых данных о принятых сотрудниках->Сотрудник принят на работу | Окно, в котором нового принятого сотрудника заносят в базу данных уже имеющихся сотрудников и указывают всю информацию о нем |
| Сотрудник принят на работу->Составление нового расписания->Изменение расписания->Распределение должностей по новому расписанию->Сотрудник выходит на работу | Окно составления и редактирования рабочего графика |
| Сотрудник принят на работу->Составление нового расписания-> увольнение сотрудников-> написание письма об увольнении сотрудником->Расторжение трудового договора->Сотрудник уволился | Окно для удаления из БД уволенных сотрудников |
| Менеджер отдела кадров | Ведёт составление рабочего графика, заносит в БД информацию о сотрудниках и занимается распределением зарплаты |
| Расписание | Выходной документ, формирующийся в результате работы менеджера отдела кадров |
| Клиент | Может просматривать общую информацию о товарах, покупать товары и отзываться на свободную вакансию на работу в магазине |
| Анкета для собеседования | Входной документ, входящий в процесс найма соискателя на работу |
| Юрист | Заносит информацию о договорах в БД |

Остальные окна дизайн проекта информационной системы указываются в приложении Г.

## **1.4. Эксплуатационное назначение разработки**

### **1.4.1. Роли**

Описание ролей информационной системы «Гостиница», выявленных в методологии EPC (Таблица 1.7)

Таблица 1.7. Описание ролей информационной системы

|  |  |
| --- | --- |
| Название роли | Описание роли |
| Продавец консультант | * Оформление лицензий * Занесение товаров в книгу продаж * Редактирование информации о проданных товарах * Консультация для клиента с целью продажи товара |
| Юрист магазина | * Подписание договоров с инвесторами и поставщиками * Решение юридических вопросов магазина |
| Менеджер отдела кадров | * Оформление нового сотрудника * Изменение данных о повышении квалификации * Повышение сотрудника * Понижение сотрудника * Увольнение сотрудника * Изменение личных данных сотрудника   Создание аккаунта новому сотруднику |
| Администратор | * Редактирование в БД информации о всех товарах в магазине * Редактирование рабочего расписания * Подписание трудовых договоров * Списание товара со склада * Согласование условий договоров с поставщиком * Распределение должностей между сотрудниками |
| Гость | * Имеет доступ к просмотру общей информации о товарах * Имеет доступ к просмотру свободных вакансий |

### **1.4.2. Требования к параметрам технических средств**

Описание минимальных требований к аппаратным и программным средствам для успешной эксплуатации ИС, со стороны двухзвенной клиент – серверной архитектуры, где в качестве элементов архитектуры выступают клиентский персональный компьютер (таблица 1.8, таблица 1.9) и серверная установка (таблица 1.10, таблица 1.11).

Таблица 1.8 Минимальные аппаратные требования к персональному компьютеру пользователя

|  |  |
| --- | --- |
| Название элемента | Описание |
| CPU | [**Процессор** Intel Pentium Clarkdale](https://market.yandex.ru/product--protsessor-intel-pentium-clarkdale/6088083?show-uid=15820444677232002213516004&nid=55330&lr=213&text=%D0%BF%D1%80%D0%BE%D1%86%D0%B5%D1%81%D1%81%D0%BE%D1%80%20itanium%20%D1%86%D0%B5%D0%BD%D0%B0&context=search) или выше |
| RAM | Не менее 1 Гб |
| HDD/SSD | Кол-во требуемой памяти 50 Мб |
| Видеокарта | 512 МБ видеопамяти GDDR3 и выше.  частота ядра/памяти: 589/1000 МГц и выше. |
| Сетевая карта | скорость 10/100/1000 Мбит/с и выше.  1 разъем RJ-45 |
| Манипуляторы | Компьютерная мышь – любая |
| Периферийные устройства | Монитор – любой с  разрешением 1024x768 (4:3) и выше,  Клавиатура – (любая с интерфейсом USB) |

Таблица 1.9 Минимальные программные требования к персональному компьютеру пользователя

|  |  |
| --- | --- |
| Название элемента | Описание |
| Операционная система | Windows XP и выше |
| Дополнительное ПО | Internet explorer, Приложение «Магазин оружия» |

Таблица 1.10 Минимальные аппаратные требования к серверной установке

|  |  |
| --- | --- |
| Название элемента | Описание |
| CPU | Intel (или совместимый) 2000 МГц или выше |
| RAM | 4 ГБ или более |
| HDD/SSD Raid | 20 ГБ или более |
| Сетевая карта | Скорость 10/100/1000 Мбит/с и выше.  1 разъем RJ-45 |

Таблица 1.11 Минимальные программные требования к серверной установке

|  |  |
| --- | --- |
| Название элемента | Описание |
| Операционная система | Поддерживаемые ОС:   * Windows Vista SP2, * Windows Server 2008 SP2, * Windows 7 SP1, * Windows Server 2008 R2 SP1, * Windows 8, * Windows Server 2012, * Windows 10. |
| Система управления базами данных | MS SQL Server 2012 и выше |

# **2. СПЕЦИАЛЬНАЯ ЧАСТЬ**

## **2.1. Постановка задачи**

В соответствии с бланком задания, на котором предоставлены требования к проекту, нужно переформулировать в соответствии с выбранной предметной областью и подсистемами, в виде маркированного списка.

* Проведение анализа магазина по методологиям IDEF0, DFD, EPC;
* Разрабатывать тесты к разрабатываемой информационной системе магазина;
* Реализовать стандартные функций БД магазина;
* Автоматизировать бизнес процессы подсистем магазина;
* Реализовать в БД магазина оружия объекты (Triggers, Procedures, View, Roles);
* Создать интуитивно – понятный интерфейс информационной системы магазина;
* Провести адаптацию разработанной информационной системы магазина;
* Разработать механизм аутентификации пользователей в информационной системе магазина;
* Провести оценку качества разработанной информационной системы «Магазин оружия»;
* Разработать техническую сопроводительную документацию;
* Реализовать механизм внедрения информационной системы «Магазин».

## **2.2. Проектирование**

### **2.2.1. Внутренняя спецификация**

#### **2.2.1.1. Описание входных и выходных данных**

Входные данные, полученные в результате проведения анализа предметной области по методологиям DFD и EPC (Таблицы 2.1,2.2)

Таблица 2.1 Входные данные

|  |  |  |  |
| --- | --- | --- | --- |
| Логическое название | Физическое название | Тип данных | Описание |
| Сотрудник | | | |
| Фамилия сотрудника | Surname\_Employee | VARCHAR(30) | NOT NULL |
| Имя сотрудника | Name\_Employee | VARCHAR(30) | NOT NULL |
| Отчество сотрудника | Middle\_Name\_Employee | VARCHAR(30) | NOT NULL |
| Стаж | Job\_Expirience | VARCHAR(10) | NOT NULL |
| Дата трудоустройства | Empoyment\_Data | VARCHAR(10) | NOT NULL маска – 11.11.1111 |
| Лицензия | | | |
| Номер лицензии | License\_Number | VARCHAR(30) | NOT NULL |
| Накладная | | | |
| Номер накладной | Number\_Nakladnaya | VARCHAR(30) | NOT NULL |
| Договор | | | |
| Номер договора | Number\_Dogovor | VARCHAR(30) | NOT NULL |
| Должность | | | |
| Название должности | Name\_Position | VARCHAR(30) | NOT NULL |
| Зарплата | Salary | VARCHAR(30) | NOT NULL |
| Авторизация | | | |
| Логин | Login | VARCHAR(30) | NOT NULL  Длина равна или больше 8 символов  Уникальный |
| Пароль | Password | VARCHAR(30) | NOT NULL  Присутствуют в обязательном порядке хотя бы одна цифра от 0 до 9, любые буквы английского алфавита и один спецсимвол, длина пароля равна или больше 8 символов. |
| Клиент | | | |
| Фамилия клиента | Surname\_Klient | VARCHAR(30) | NOT NULL |
| Имя клиента | Name\_Klient | VARCHAR(30) | NOT NULL |
| Отчество клиента | Middle\_Name\_Klient | VARCHAR(30) | NOT NULL |
| Номер телефона | Telephone\_Number | VARCHAR(16) | NOT NULL  Маска  +7-999-999-99-99 |
| Тип лицензии | | | |
| Название типа лицензии | License\_Number | VARCHAR(20) | NOT NULL |
| Оружие | | | |
| Название оружия | Name\_Weapon | VARCHAR(30) | NOT NULL |
| Точность | Accuracy | VARCHAR(30) | NULL |
| Скорострельность | Fire\_Rate | VARCHAR(30) | NULL |
| Патронов в магазине | Shells\_In\_Store | VARCHAR(3) | NULL |
| Цена | Cost | VARCHAR(30) | NOT NULL |
| Тип оружия | | | |
| Название типа оружия | Name\_Type\_Weapon | VARCHAR(30) | NOT NULL |
| Поставщик | | | |
| Название организации | Name\_Organization | VARCHAR(30) | NOT NULL |
| Поставка | | | |
| Дата | Date | VARCHAR(30) | NOT NULL маска – 11.11.1111 |
| Кол-во принятых оружий | Ammount\_Accepted  \_Weapon | INT | NOT NULL |
| Кол-во принятых модификаций | Ammount\_Accepted\_  Modifications | INT | NOT NULL |
| Кол-во принятых патронов | Ammount\_Accepted  \_Ammo | INT | NOT NULL |
| Роль | | | |
| Заголовок роли | Title\_Role | VARCHAR(30) | NOT NULL |
| Роль клиента | Klient | INT | NOT NULL |
| Роль сотрудника | Employee | INT | NOT NULL |
| Роль админа | Admin | INT | NOT NULL |

Таблица 2.2 Выходные данные

|  |  |  |
| --- | --- | --- |
| Название документа | Поля документа | Формат вывода |
| Расписание | Номер сотрудника | \*xlsx |
| Рабочие дни |
| Рабочие часы |
| Накладная | Номер накладной | \*docx, \*pdf |
| Название организации-поставщика |
| Подпись ответственного лица со стороны магазина |
| Подпись ответственного лица со стороны организатора |
| Название товара |
| Количество товара |
| Дата разгрузки |
| Договор | Номер договора | \*docx, \*pdf |
| Название организации-поставщика |
| Подпись ответственного лица со стороны магазина |
| Подпись ответственного лица со стороны организатора |
| Дата заключения |
| Срок действия |
| Предмет договора |
| Статус договора |

#### **2.2.1.2. Модель данных**

Иллюстрация логической модели данных;

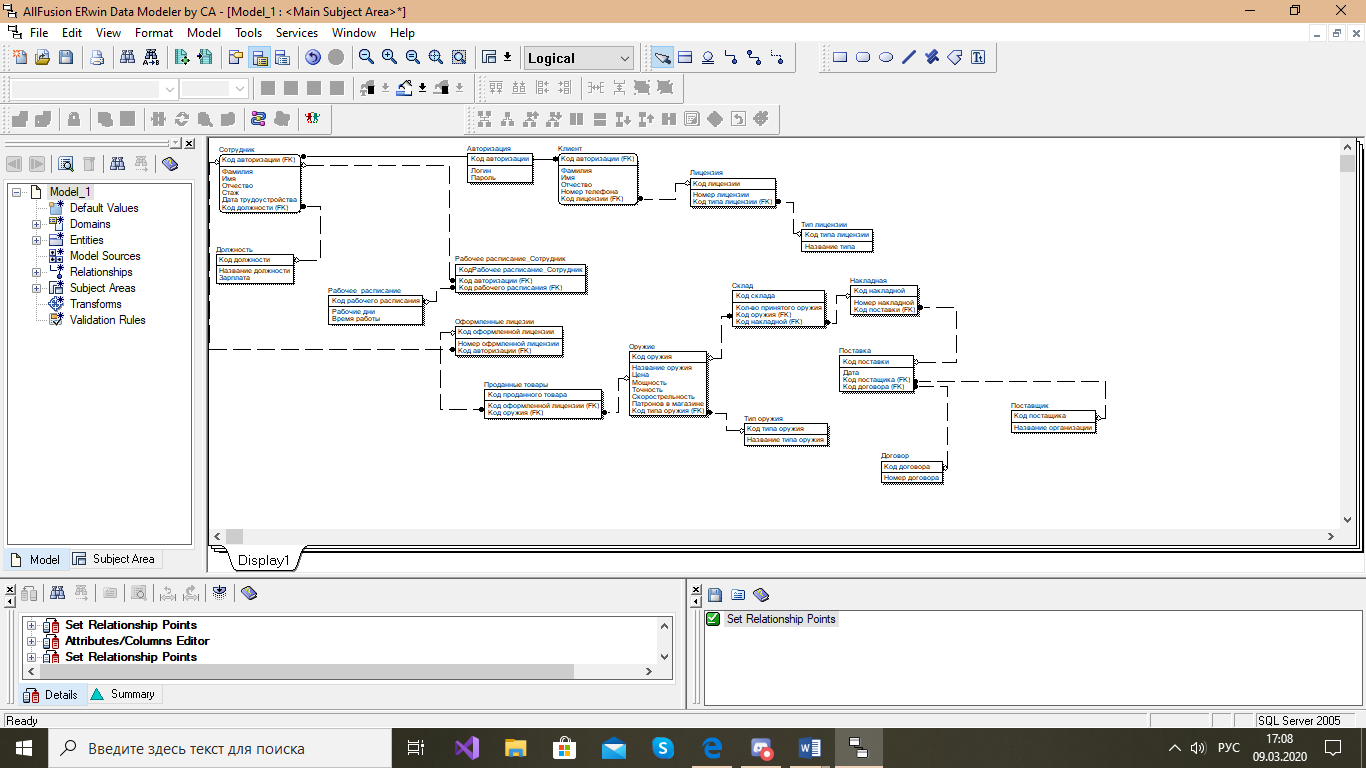


Рисунок 2.1 – Логическая модель

Иллюстрация Физической модели данных;

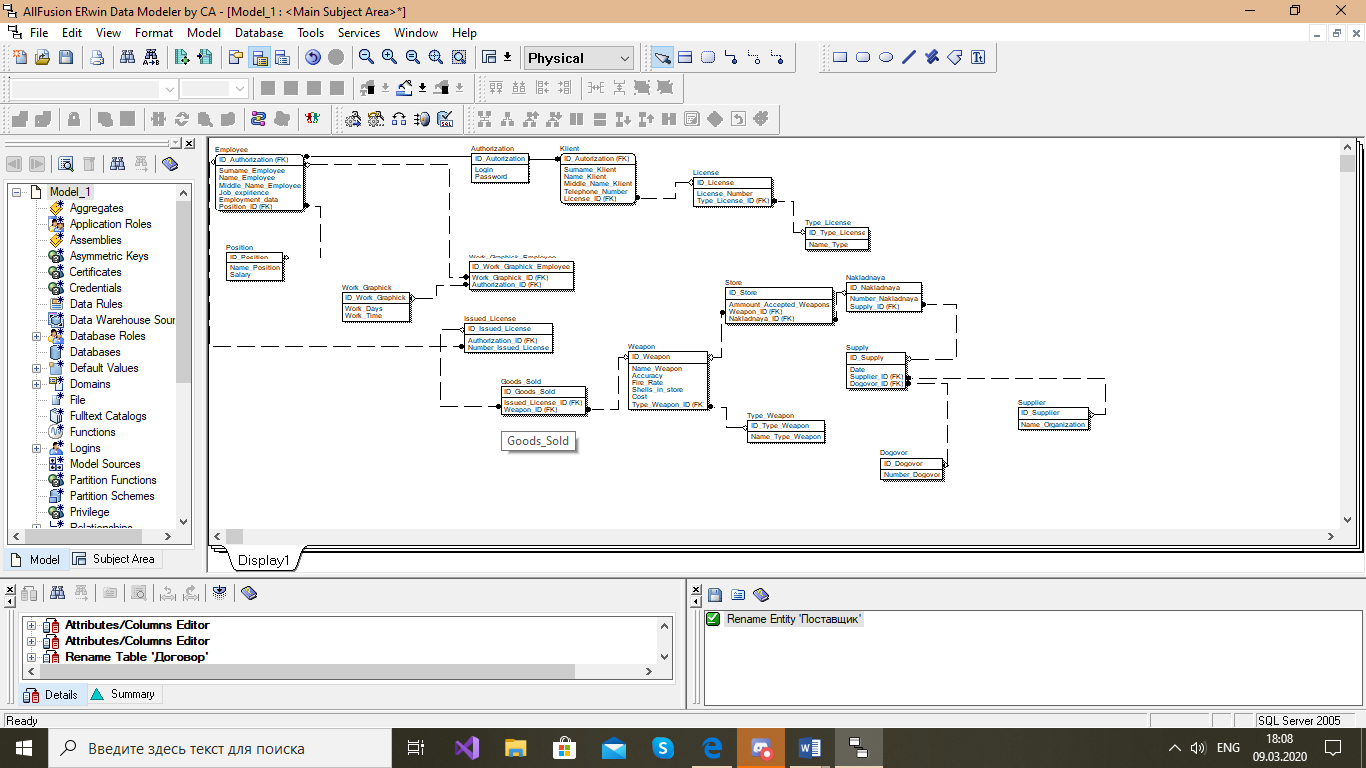


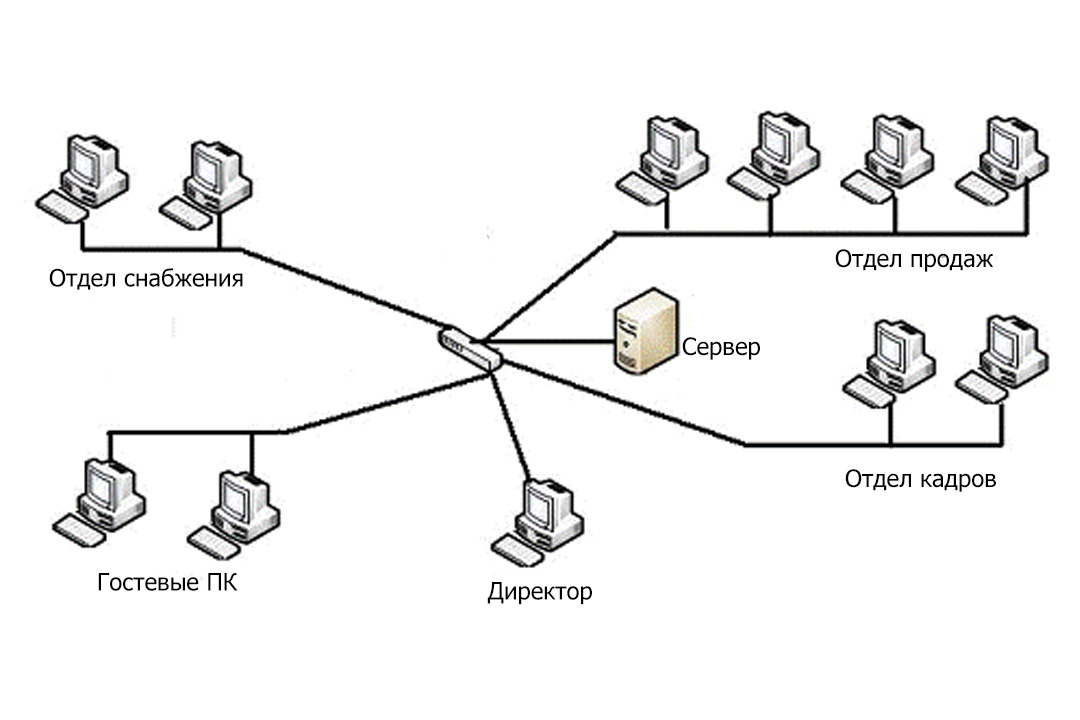
Рисунок 2.2 – Физическая модель

Подробное описание таблиц структуры данных будет указано в приложении Д.

#### **2.2.1.3. Архитектура информационной системы**

Иллюстрация двухзвенной архитектуры ИС на территории предприятия. (Рисунок 1.13)

Рисунок 2.3 – Топология сети



### **2.2.2. Внешняя спецификация**

### **2.2.2.1. Функциональная схема (Рисунок 2.4)**

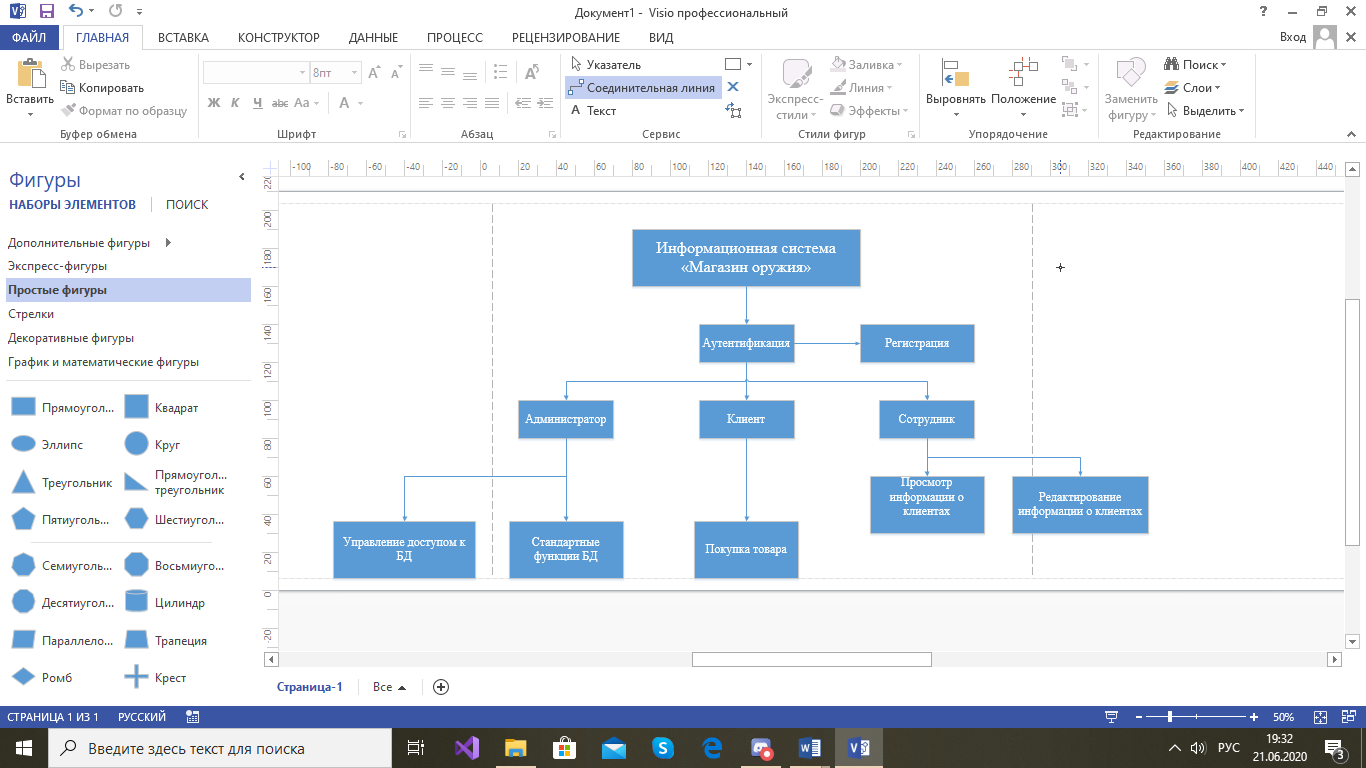
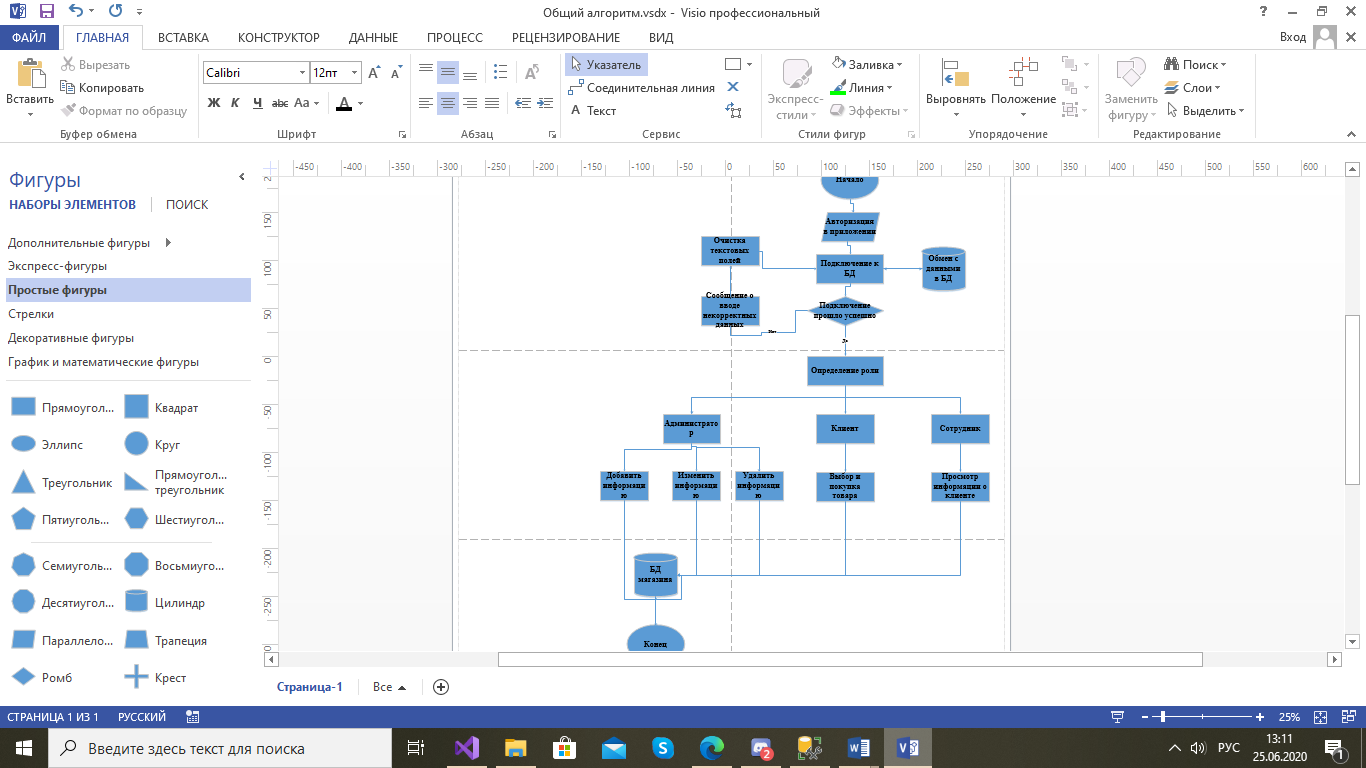


Рисунок 2.4 - Функциональная схема

### **2.2.2.2. Алгоритмы**

* Общий алгоритм «Магазин оружия»



* Алгоритм бизнес процесса «Покупка товара» (Рисунок 2.5)

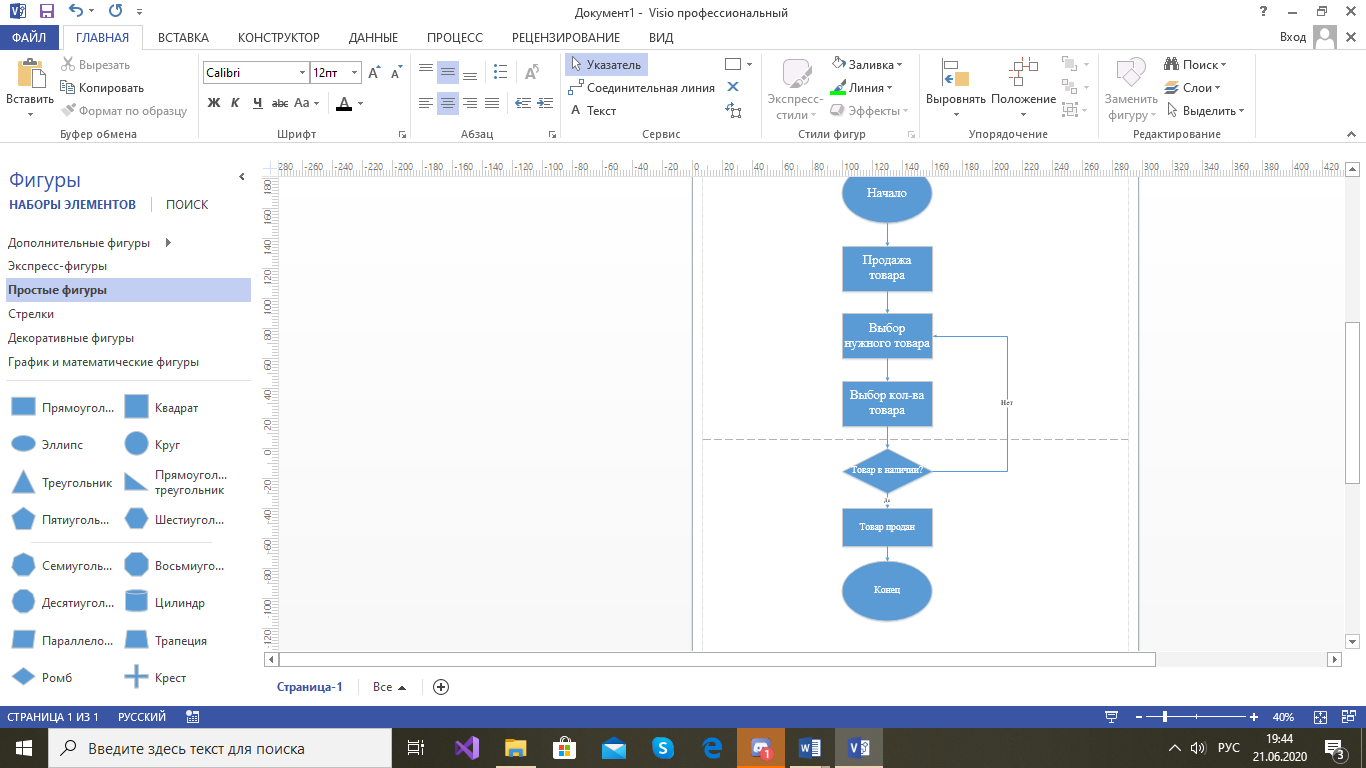


Рисунок 2.5 – Продажа товара

* Алгоритм бизнес процесса «Учет товаров» (Рисунок 2.6)

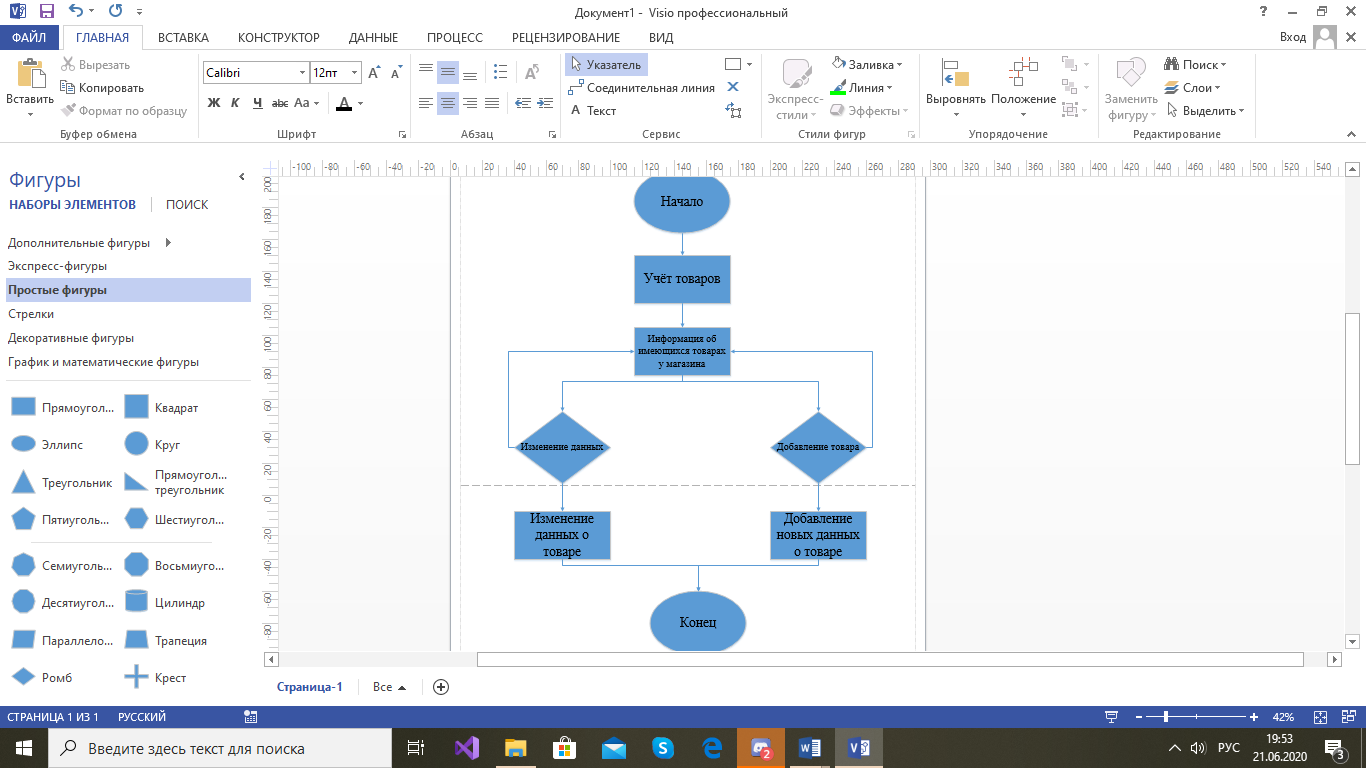


Рисунок 2.6 – Учёт товаров

* Алгоритм бизнес процесса «Добавление товаров» (Рисунок 2.7)

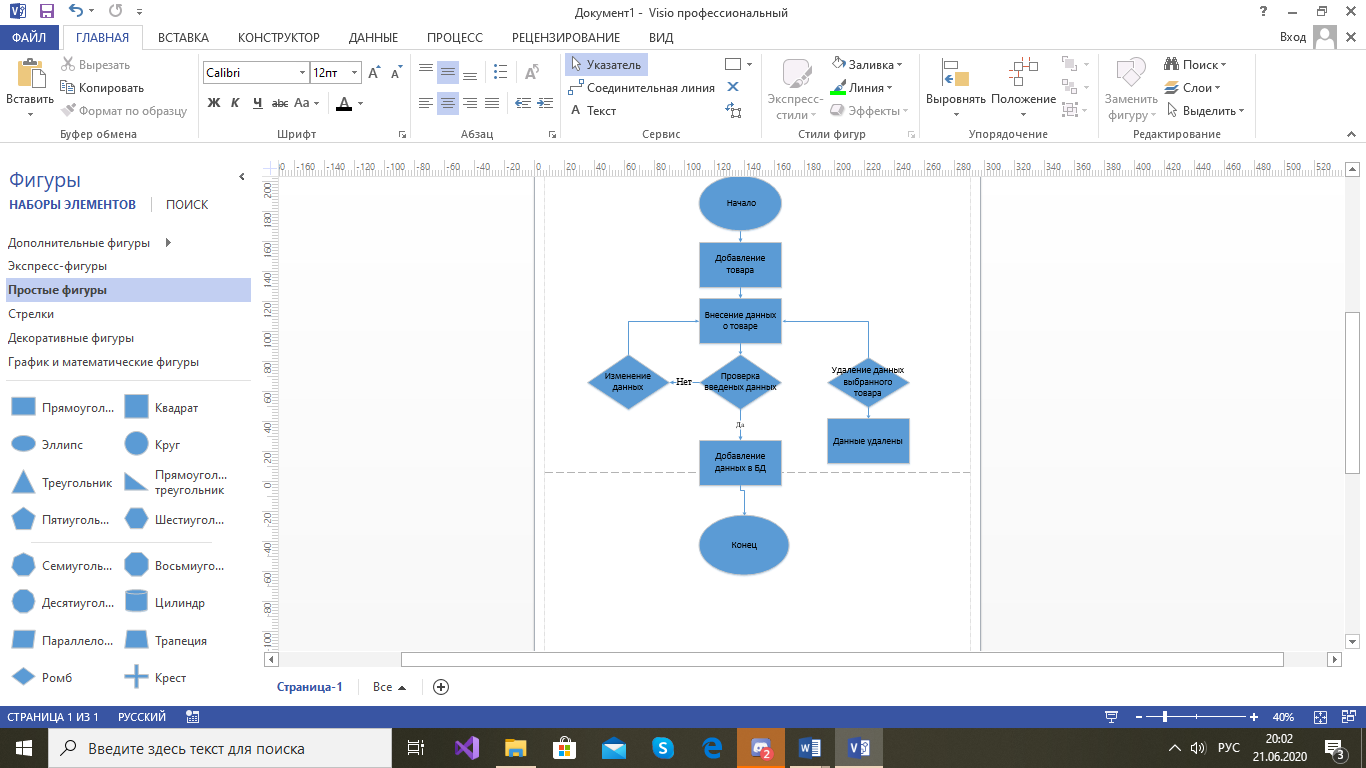


Рисунок 2.7 – добавление товара

* Алгоритм бизнес процесса «Список клиентов» (Рисунок 2.8)

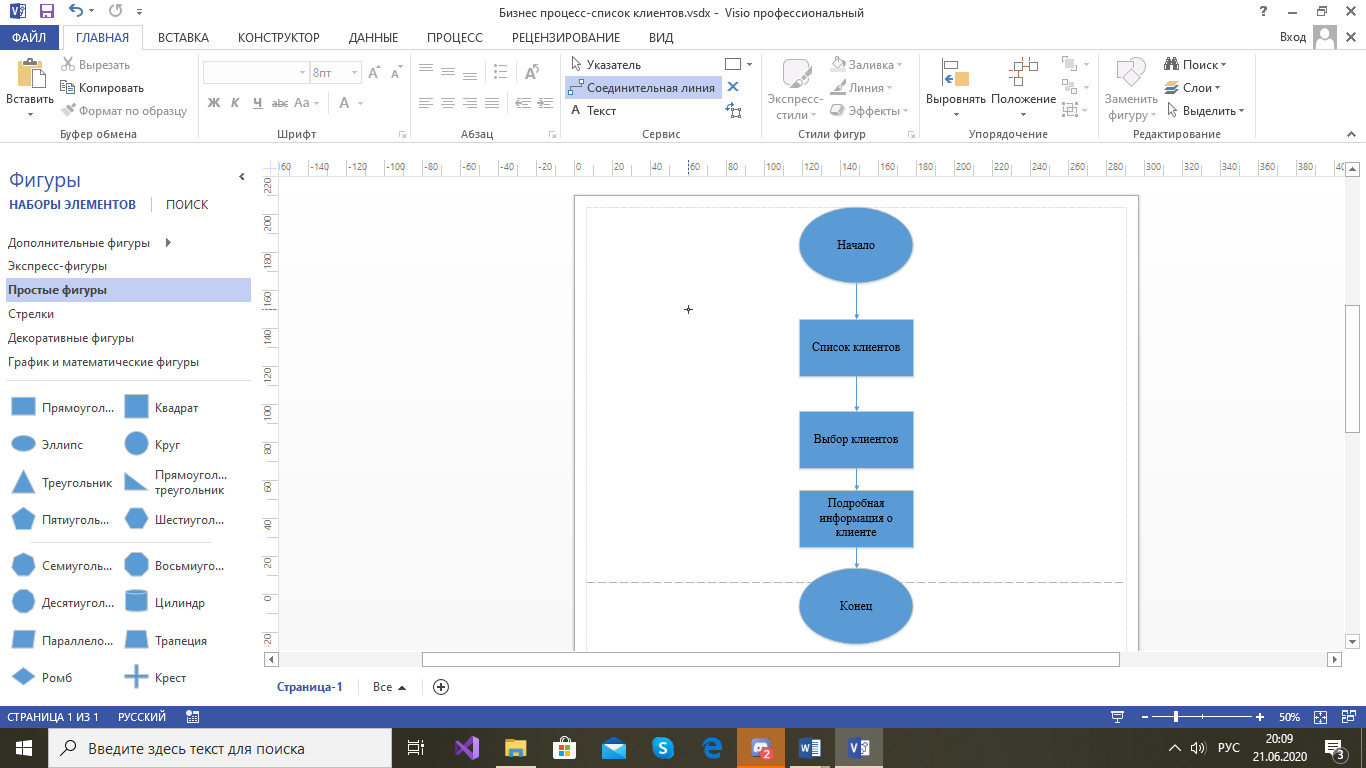


Рисунок 2.8 – Список клиентов

* Алгоритм бизнес процесса «Приход поставок» (Рисунок 2.9)

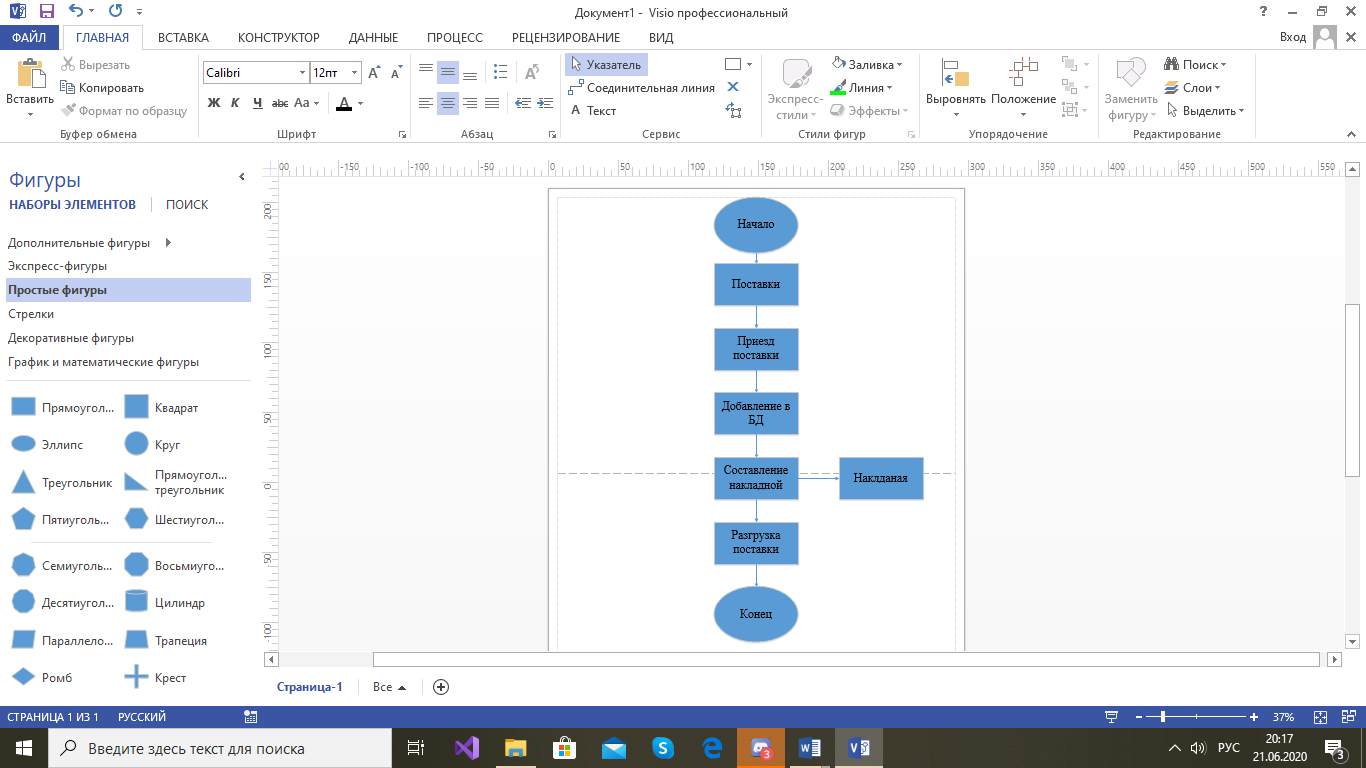


Рисунок 2.9 – Приход поставок

* Алгоритм бизнес процесса «Регистрация пользователя» (Рисунок 2.10)

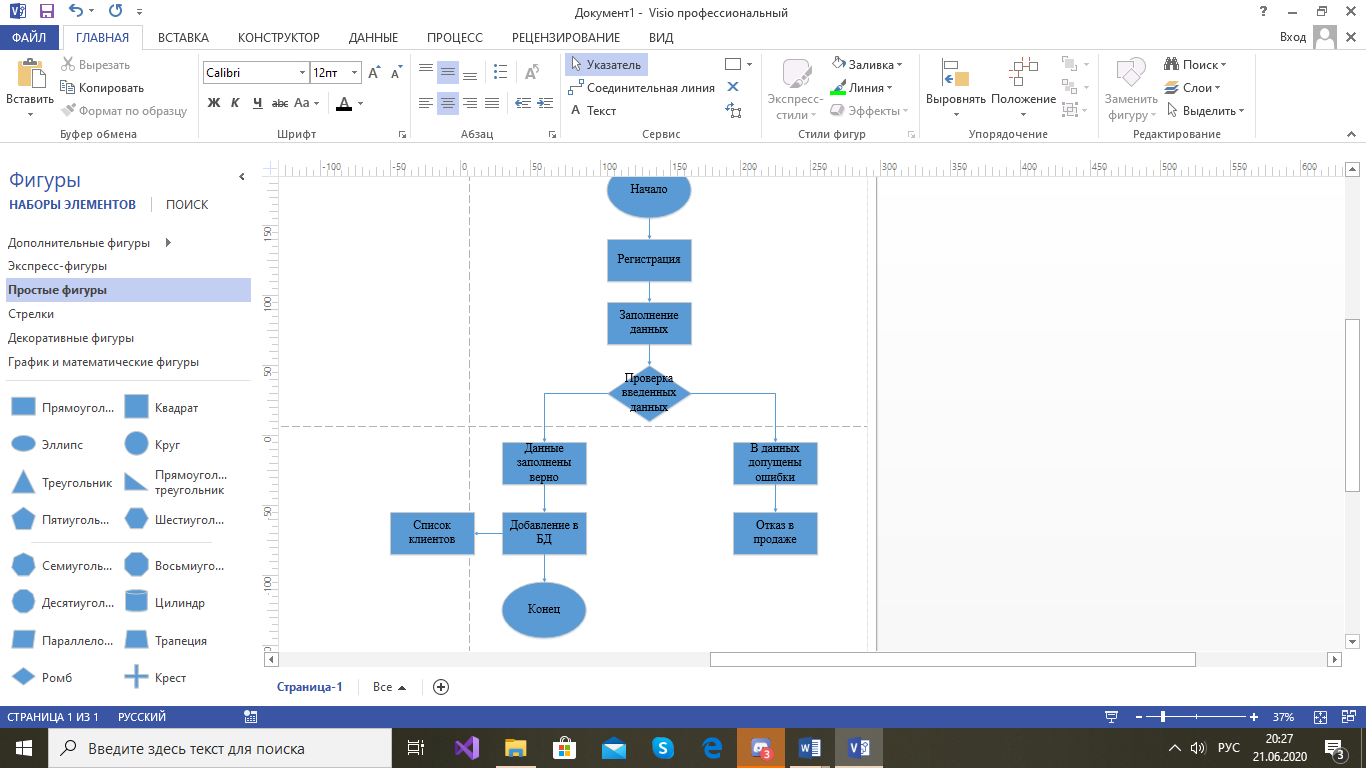


Рисунок 2.10- Регистрация пользователя

### **2.2.2.3. Шифрование данных (Рисунок 2.11)**

Криптосистема RSA используется в самых различных продуктах, на различных платформах и во многих отраслях. В настоящее время криптосистема RSA встраивается во многие коммерческие продукты, число которых постоянно увеличивается. Также ее используют операционные системы Microsoft, Apple, Sun и Novell. В аппаратном исполнении RSA алгоритм применяется в защищенных телефонах, на сетевых платах Ethernet, на смарт-картах, широко используется в криптографическом оборудовании. Кроме того, алгоритм входит в состав всех основных протоколов для защищенных коммуникаций Internet, в том числе S/MIME, SSL и S/WAN, а также используется во многих учреждениях, например, в правительственных службах, в большинстве корпораций, в государственных лабораториях и университетах.

Алгоритм RSA, можно считать наиболее распространенной криптосистемой открытого (public) ключа в мире и это количество имеет явную тенденцию к увеличению по мере роста Internet.

Главное достоинство алгоритма RSA - имея открытый ключ и зная алгоритм шифрования невозможно повторить закодированное сообщение, на базе алгоритма RSA работает программа шифрования PGP, реализуются хэш-функции (элетронно-цифровая подпись).

Хотя из-за низкой скорости шифрования (около 30 кбит/с при 512 битном ключе на процессоре 2 ГГц), сообщения обычно шифруют с помощью более производительных симметричных алгоритмов со случайным ключом (сеансовый ключ), а с помощью RSA шифруют лишь этот ключ. Такой механизм имеет потенциальные уязвимости ввиду необходимости использовать криптостойкий генератор случайных чисел для формирования случайного сеансового ключа симметричного шифрования и эффективно противостоящий атакам симметричный криптоалгоритм (в данное время широкое применение находят AES, IDEA, Serpent, Twofish).

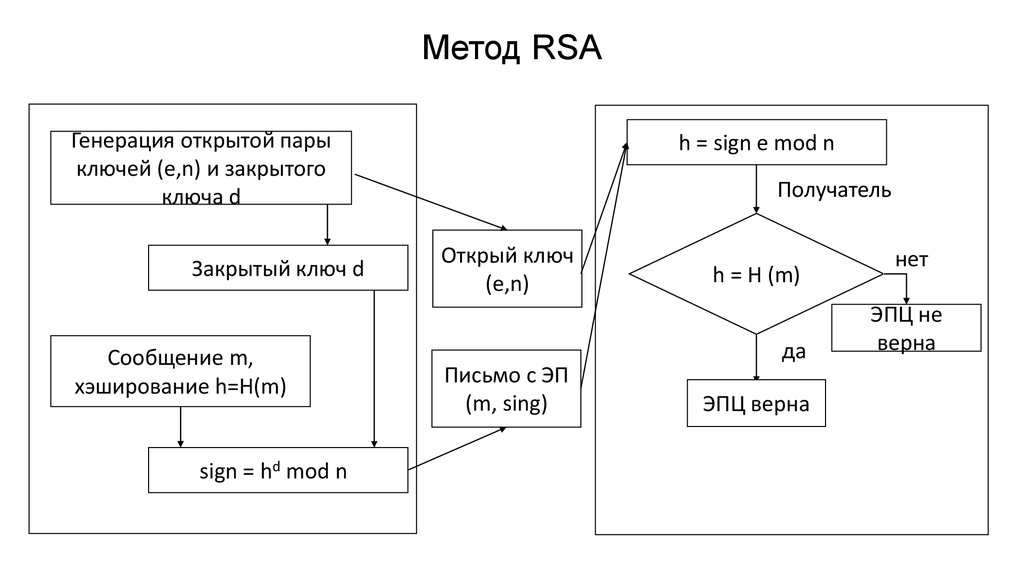


Рисунок 2.11 - метод шифрования "RSA"

### **2.2.2.4. Целостность данных**

Таблица 2.3 - целостность данных

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Поле контроля | Ситуации | | | Реакция программы |
| а | б | в |
| Фамилия | Ввод символов (Temchenko!) | Пользователь оставит поле пустым | Ввод имени русскими символами (Темченко) | Ситуация А – Программа не даст вписать символы |
| Ситуация Б – Приложение выдаст окно что данные не введены |
| Ситуация В – Приложение успешно запишет данные |
| Имя | Ввод символов (Денис!) | Пользователь оставит поле пустым | Ввод имени русскими символами (Денис) | Ситуация А – Приложение не даст вписать символы |
| Ситуация Б – Приложение выдаст окно что данные не введены |
| Ситуация В – Приложение успешно запишет данные |
| Отчество | Ввод символов (Igorevich!) | Пользователь оставит поле пустым | Ввод имени русскими символами (Игоревич) | Ситуация А – Приложение не даст вписать символы |
| Ситуация Б – Приложение выдаст окно что данные не введены |
| Ситуация В – Приложение успешно запишет данные |
| Опыт работы | Ввод символов  (18 лет) | Пользователь оставит поле пустым | Размер записываемого поля превысит значение | Ситуация В – Приложение успешно запишет данные |
| Ситуация Б – Приложение выдаст окно что данные не введены |
| Ситуация В – Приложение выдаст ошибку |
| Дата трудоустройства | Ввод символов  (20 июня 20 года) | Пользователь оставил поле пустым | Верный формат ввода (20.06.2020) | Ситуация А - Приложение выдаст ошибку |
| Ситуация Б – Приложение выдаст окно что данные не введены |
| Ситуация В – Приложение успешно запишет данные |
| Название должности | Пользователь превысил размер поля | Пользователь оставит поле пустым | Верный формат ввода (Кассир) | Ситуация А - Приложение выдаст ошибку |
| Ситуация Б - Приложение выдаст окно что данные не введены |
| Ситуация В – Приложение успешно запишет данные |
| Пароль | Пользователь не вписал цифру, заглавную латинскую букву и пароль будет меньше 8 символов (zigchik) | Пользователь оставит поле пустым | Верный формат ввода (@Qwerty1234567890!!!) | Ситуация А – Данные не введутся |
| Ситуация Б - Приложение выдаст окно что данные не введены |
| Ситуация В – Приложение успешно запишет данные |
| Повтор пароля | Пользователь не вписал цифру, заглавную латинскую букву и пароль будет меньше 8 символов (zigchik) | Пароли не совпадают (@!!!Qwerty1)  (Qwtyr1!!!) | Верный формат ввода (@Qwerty1234567890!!!) | Ситуация А – Данные не введутся |
| Ситуация Б - Данные не введутся |
| Ситуация В – Приложение успешно запишет данные |
| Логин | Логин будет меньше 8 символов (Zigchik) | Пользователь оставит поле пустым | Верный формат ввода (Zigchik Zigchik) | Ситуация А – Приложение не даст вписать короткий логин |
| Ситуация Б - Приложение выдаст окно что данные не введены |
| Ситуация В – Программа успешно запишет данные |

### **2.2.2.5. Структурная схема (Рисунок 2.12)**

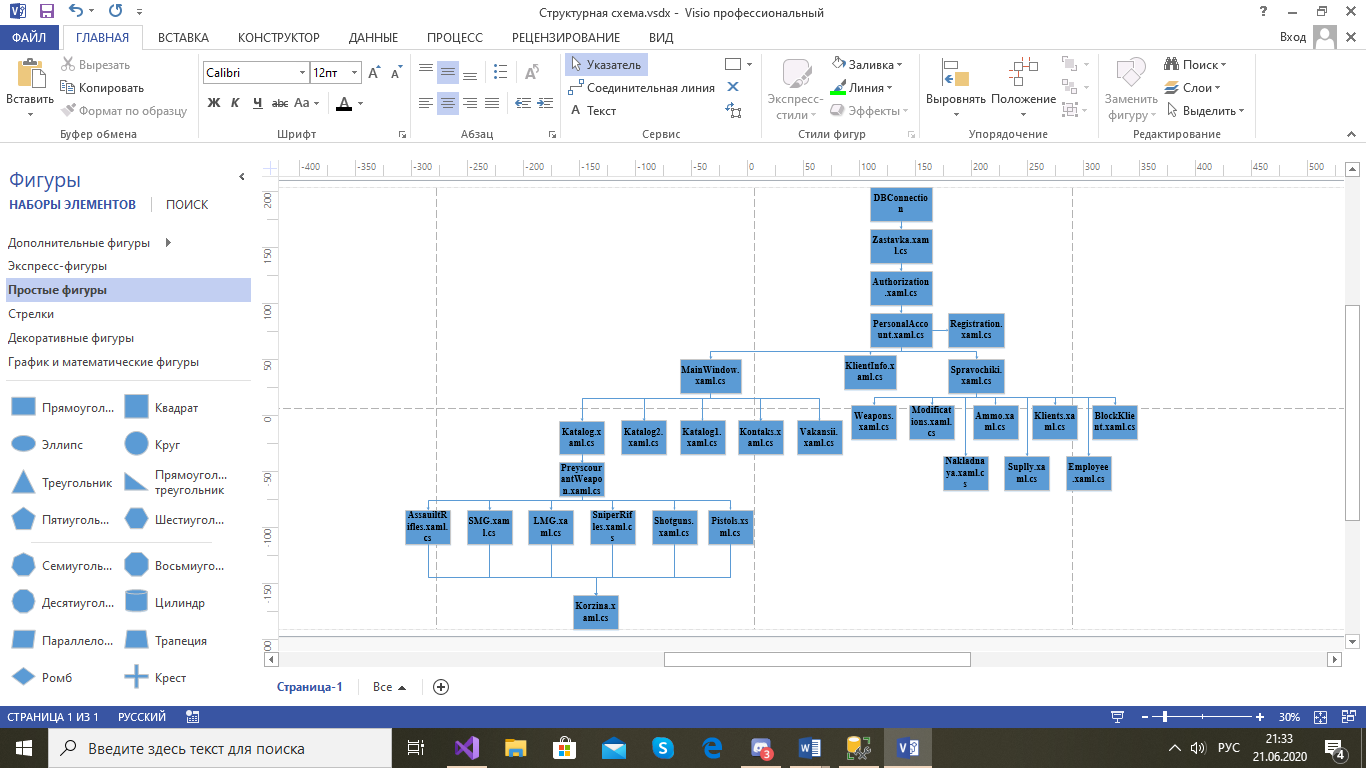


Рисунок 2.12 – Структурная схема

### **2.2.2.6. Схема пользовательского интерфейса (Рисунок 2.13)**

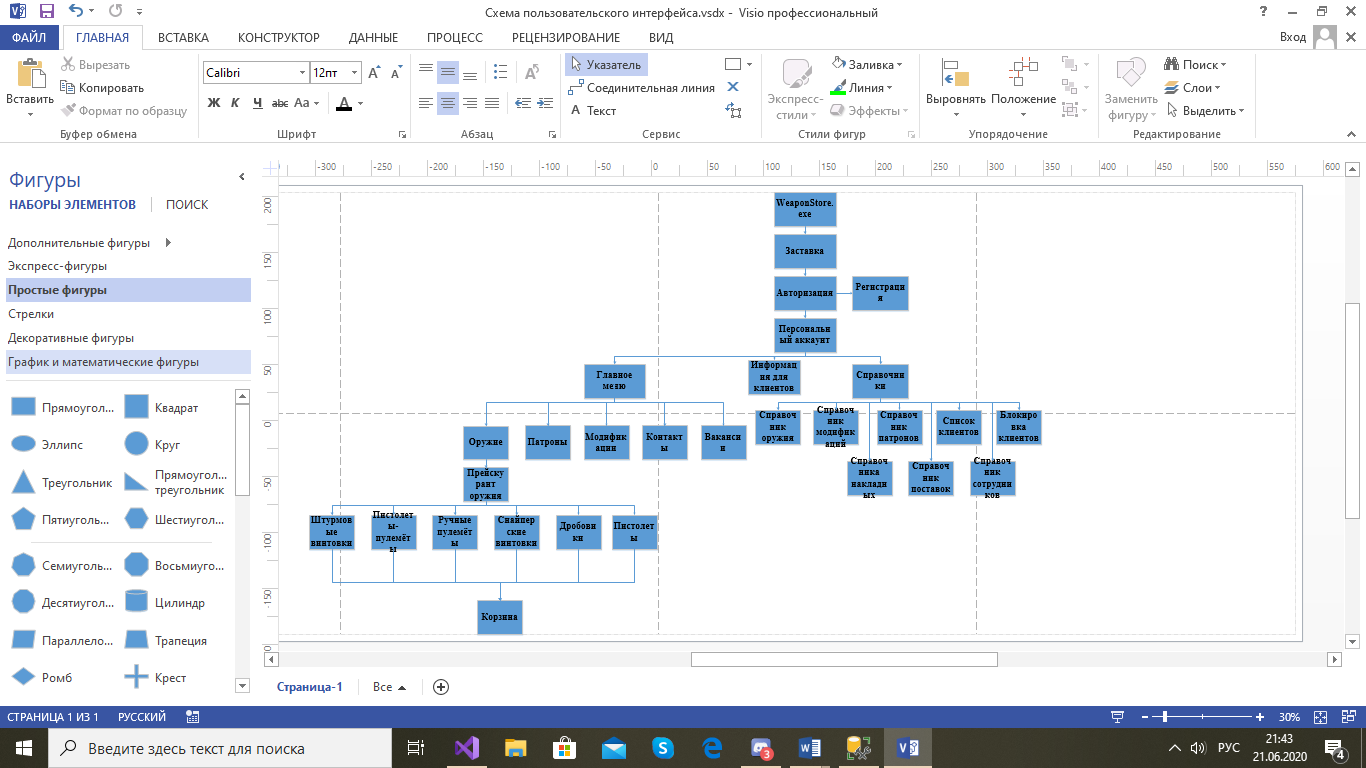


Рисунок 2.13 – Схема пользовательского интерфейса

### **2.2.2.7. Тесты**

Таблица 2.4 – Тестирование на пустые значения вкладки «Справочник оружия» добавить

|  |  |
| --- | --- |
| Наименование проекта | WeaponStore |
| Номер версия | 2.0 |
| Test Case1 | Темченко |
| Дата тестирования | 20.06.20 |
| Приоритет тестирования  (Мал/Сред/Высок) | Малый |
| Описание тестирования | Проверка на добавление пустых значений «Справочник оружия» |
| Ход тестирования | 1. Оставить поля для ввода пустыми и выпадающие списки не выбранными 2. Нажать кнопку добавить |
| Данные тестирования | Информация об оружии: null;  Точность: null;  Скорострельность: null;  Патронов в магазине: null  Кол-во оружия на складе: null  Цена: null |
| Ожидаемый результат | Если администратор не введёт значения и не выберет данные из ListBox , то приложение выдаст окно «Error» |
| Предпосылки | 1. Запустить приложение и авторизоваться в нем, с помощью данных  2. Далее в меню выбрать «Справочник оружия» |

Таблица 2.5 – Тестирование на спецсимволы вкладки «Справочник оружия» изменить

|  |  |
| --- | --- |
| Наименование проекта | WeaponStore |
| Номер версия | 2.0 |
| Test Case2 | Темченко |
| Дата тестирования | 20.06.20 |
| Приоритет тестирования  (Мал/Сред/Высок) | Малый |
| Описание тестирования | Проверка на введение символов «Справочник оружия» |
| Ход тестирования | 1. Ввести в поле гос. точность спецсимвол 2. Нажать кнопку изменить |
| Данные тестирования | Точность: Low!; |
| Ожидаемый результат | Если администратор введет спецсимвол, то приложение выдаст окно «Error» |
| Предпосылки | 1. Запустить приложение и авторизоваться в нем, с помощью данных  2. Далее в меню выбрать «Справочник оружия» |

Таблица 2.6 – Тестирование на не выбранное значение поля таблицы «Справочник оружия» удалить

|  |  |
| --- | --- |
| Наименование проекта | WeaponStore |
| Номер версия | 2.0 |
| Test Case3 | Темченко |
| Дата тестирования | 20.06.20 |
| Приоритет тестирования  (Мал/Сред/Высок) | Малый |
| Описание тестирования | Проверка на удаление пустого значения |
| Ход тестирования | 1. Оставить поля для ввода пустыми 2. Нажать кнопку удалить |
| Данные тестирования | Информация об оружии: null;  Точность: null;  Скорострельность: null;  Патронов в магазине: null  Кол-во оружия на складе: null  Цена: null |
| Ожидаемый результат | Если администратор попытается удалить пустое значение, то приложение выдаст окно «Error» |
| Предпосылки | 1. Запустить приложение и авторизоваться в нем, с помощью данных  2. Далее в меню выбрать «Справочник оружия» |

Таблица 2.7 – Тестирование на пустое значение поля таблицы «Справочник модификаций» добавить

|  |  |
| --- | --- |
| Наименование проекта | WeaponStore |
| Номер версия | 2.0 |
| Test Case4 | Темченко |
| Дата тестирования | 20.06.20 |
| Приоритет тестирования  (Мал/Сред/Высок) | Малый |
| Описание тестирования | Проверка на добавление пустых значений «Справочник модификаций» |
| Ход тестирования | 1. Оставить поля для ввода пустыми 2. Нажать кнопку добавить |
| Данные тестирования | Название модификации: null;  Кол-во модификаций на складе: null;  Цена: null |
| Ожидаемый результат | Если администратор попытается добавить пустое значение, то приложение выдаст окно «Error» |
| Предпосылки | 1. Запустить приложение и авторизоваться в нем, с помощью данных  2. Далее в меню выбрать «Справочник модификаций» |

Таблица 2.8 – Тестирование на пустое значение поля таблицы «Справочник модификаций» изменить

|  |  |
| --- | --- |
| Наименование проекта | WeaponStore |
| Номер версия | 2.0 |
| Test Case5 | Темченко |
| Дата тестирования | 20.06.20 |
| Приоритет тестирования  (Мал/Сред/Высок) | Малый |
| Описание тестирования | Проверка на изменение пустых значений «Справочник модификаций» |
| Ход тестирования | * + - 1. Оставить поля для ввода пустыми       2. Нажать кнопку изменить |
| Данные тестирования | Название модификации: null;  Кол-во модификаций на складе: null;  Цена: null |
| Ожидаемый результат | Если менеджер попытается изменить пустые значения, то приложение выдаст окно «Error» |
| Предпосылки | 1. Запустить приложение и авторизоваться в нем, с помощью данных  2. Далее в меню выбрать «Справочник модификаций» |

Таблица 2.9 – Тестирование на не выбранное поле таблицы «Справочник модификаций» удалить

|  |  |
| --- | --- |
| Наименование проекта | WeaponStore |
| Номер версия | 2.0 |
| Test Case6 | Темченко |
| Дата тестирования | 20.06.20 |
| Приоритет тестирования  (Мал/Сред/Высок) | Малый |
| Описание тестирования | Проверка на удаление не выбранного поля |
| Ход тестирования | 1. Открыть вкладку «Справочник модификаций» 2. Нажать кнопку удалить |
| Данные тестирования | Null |
| Ожидаемый результат | Если администратор не выберет поле, то приложение выдаст окно «Не выбрана модификация» |
| Предпосылки | 1. Запустить приложение и авторизоваться в нем, с помощью данных  2. Далее в меню выбрать «Справочник модификаций» |

# **3. ТЕХНОЛОГИЧЕСКАЯ ЧАСТЬ**

## **3.1. Средства разработки**

### **3.1.1. Средства анализа**

Для этапа анализа, использовалась методология IDEF0, DFD (Process Modeler). Преимущества продукта заключается в простом графическом интерфейсе, представление дополнительной информации и контроле корректности моделей.

Для этапа анализа использовался ARIS Express, в котором создавалась последовательность функций, управляемых событиями (EPC), преимущество продукта заключается в хорошо развитом графическом интерфейсе, поддержке мощного хранилища данных, также поддержке многопользовательской работе.

### **3.1.2. Средства проектирования**

Для разработки приложения «Магазин оружия» использовались инструментальные средства: Erwin, Process Modeler, в котором была разработана даталогическая и инфологическая модель базы данных. Преимущество продукта Erwin заключается в стандартных представлениях элементов и применение открытой архитектуры.

### **3.1.3. Средства реализации**

* Для реализации приложения была выбрана инструментальная среда разработки Visual Studio. Данные среда была выбрана потому что позволяет создавать приложения для различных ОС, мобильных платформ, веб-сайты и приложения. Отличительной особенностью встроенного редактора кода является удобная навигация, понятный интерфейс. Присутствует подсветка разметки и быстрый переход с помощью полос прокрутки. Отладка доступна для всех приложений, даже если в них используется несколько языков программирования. Можно приостановить приложение и получить значения переменных на любом этапе.

Преимущества Visual Studio:

1. Поддержка множества языков программирования;
2. Кроссплатформенность разработки;
3. Встроенный контроль за выполнением многопоточного кода;
4. Наличие бесплатной редакции Community;
5. Запись происходящего во время отладки (функция IntelliTrace);

Основные возможности:

1. Редактирование, профилирование и отладка кода;
2. Нумерация строк в редакторе;
3. Поддержка языков Visual Basic, C#, C++, Java Script, HTML5, Python;
4. Автоматическое завершение кода и подсказки;
5. Рефакторинг и реструктуризация исходных текстов программ;
6. Система скачиваемых расширений для дополнительных функций;
7. Встроенные тесты производительности;
8. Эмуляция мобильных платформ;
9. Редактируемое меню внешних инструментов.

* В качестве клиентской части программного продукта была выбрана платформа WPF - расшифровывается как Windows Presentation Foundation. Входит в состав платформы .NET. Отличная платформа для создания приложений.

Преимущества WPF:

1. Отвечает современным стандартам разработки
2. Microsoft использует ее во многих своих приложениях (Visual Studio)
3. Это более гибкая система, можно сделать больше, без написания или покупки готовых элементов управления
4. С помощью XAML можно легко создавать и редактировать GUI, позволяя разделить работу дизайнера (XAML) и программиста (C#, VB.NET и др.)
5. Привязка данных позволяет лучше разделить данные и GUI
6. Для лучшей производительности, можно использовать аппаратное ускорение при от рисовки GUI
7. Позволяет создавать GUI как для Windows приложений, так и для приложений Web (Silverlight/XBAP)

* В качестве серверной части была выбрана СУБД SQL Server Management Studio (SSMS).

**Среда SQL Server Management Studio** (SSMS) – это графический набор средств, для разработки сценариев на T-SQL и управления всеми компонентами SQL Server.

Основные достоинства языка SQL:

1. Стандартность – использование языка SQL в программах стандартизировано международными организациями;
2. Возможность переноса с одной вычислительной системы на другую – СУБД может быть ориентирована на различные вычислительные системы, однако приложения, созданные с помощью SQL, допускают использование как для локальных БД, так и для крупных многопользовательских систем;
3. Возможность создания интерактивных запросов – SQL обеспечивает пользователям немедленный доступ к данным, при этом в интерактивном режиме можно получить результат запроса за очень короткое время без написания сложной программы;
4. Возможность программного доступа к БД – язык SQL легко использовать в приложениях, которым необходимо обращаться к базам данных. Одни и те же операторы SQL употребляются как для интерактивного, так и программного доступа, поэтому части программ, содержащие обращение к БД, можно вначале проверить в интерактивном режиме, а затем встраивать в программу;
5. Возможность динамического изменения и расширения структуры БД – язык SQL позволяет манипулировать структурой БД, тем самым обеспечивая гибкость с точки зрения приспособленности БД к изменяющимся требованиям предметной области;
6. Поддержка архитектуры клиент-сервер – SQL – одно из лучших средств для реализации приложений на платформе клиент-сервер. SQL служит связующим звеном между взаимодействующей с пользователем клиентской системой и серверной системой, управляющей БД, позволяя каждой из них сосредоточиться на выполнении своих функций.

### **3.1.4. Методы отладки**

В качестве метода отладки программного обеспечения были использован метод: Метод ручного тестирования - выполнялся тест программы вручную. Все тесты представлены в разделе «Специальная часть» (Рисунки 3.1, 3.2).

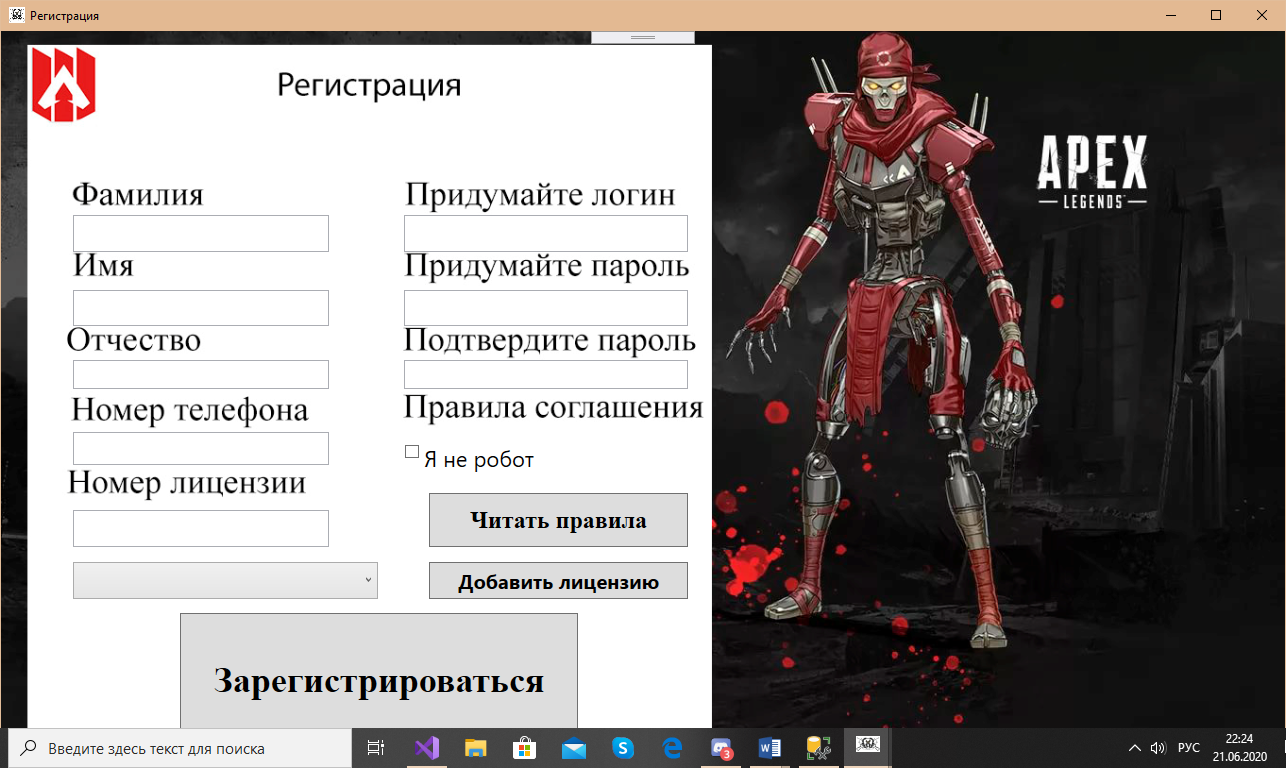


Рисунок 3.1 – Регистрация

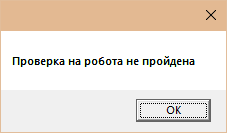


Рисунок 3.2 – Ошибка при регистрации

### **3.1.5. Методы адаптации и конфигурации**

Таблица 3.1 - методы адаптации и конфигурации

|  |  |
| --- | --- |
| Описание элемента | Код элемента |
| Адаптация на проверку подключения – если отсутствует подключение к источнику данных | static void Main()  {  bool Create\_app;  \_instanse = new Mutex(true, \_app\_Name, out Create\_app);  if (Create\_app)  {  try  {  //В случае если процесс не найден в системе  //создаётся экземпляр класса конфигурации подключения  //к источнику данных  Configuration\_class configuration  = new Configuration\_class();  //Вызывается метод получения информации из реестра  configuration.SQL\_Server\_Configuration\_Get();  //Попытка открыть поддключение к источнику данных  Configuration\_class.connection.Open();  //В случае если подключение открыто  //в переменую присваевается значение истины  connect = true;  }  catch  {  //Стилистика элементов уаправления ОС  Application.EnableVisualStyles();  //Обязательная дрянь, хер поймиёшь зачем она нужна  Application.SetCompatibleTextRenderingDefault(false);  //---НЕ СОЗДАНО--Объявление экземпляра класса  //конфигурации полдключения к источнику данных  Form1 connection = new Form1();  //---НЕ СОЗДАНО---Вывод окна подключения к источнику данных  //В режиме диалогового окна  connection.ShowDialog();  }  finally  {  //Закрытие подключения к источнику данных  Configuration\_class.connection.Close();  //Проверка состояния подключения к источнику данных  switch (connect)  {  //Если пользователь НЕ настроил подключение то  //вывод сообщения об ошибке  case false:  MessageBox.Show("Ошибка подключения " +  "к источнику данных!",  "Bruh",  MessageBoxButtons.OK,  MessageBoxIcon.Error);  //Закрытие приложения  Environment.Exit(0);  break;  //В случе если подключение установлено  case true:  try  {  //Application.EnableVisualStyles();  //Application.SetCompatibleTextRenderingDefault(false);  //Application.Run(new Form2());  }  catch  {  }  break;  }  }  } } |
| Добавление многопоточности в формировании документов – Экспорт данных выносится в отдельный метод | private void createExportDoc()  {  try  {  DBProcedures con = new DBProcedures();  var connectionKlientDetail = con.getKlientListDetail();  if (extension == string.Empty)  {  MessageBox.Show("Не выбран тип экспортруемого файла");  return;  }  switch (extension)  {  case (".docx"):  string pathDocumentDOCX = Session.baseDir + "Список клиентов" + extension;  DocX document = DocX.Create(pathDocumentDOCX);  Xceed.Document.NET.Paragraph paragraph = document.InsertParagraph();  paragraph.  AppendLine("Документ '" + "Отчет список клиентов" + "' создан " + DateTime.Now.ToShortDateString()).  Font("Time New Roman").  FontSize(16).Bold().Alignment = Alignment.left;  paragraph.AppendLine();  Xceed.Document.NET.Table doctable = document.AddTable(connectionKlientDetail.Count + 1, 2);  doctable.Design = TableDesign.TableGrid;  doctable.TableCaption = "Список клиентов";  doctable.Rows[0].Cells[0].Paragraphs[0].Append("Список клиентов").Font("Times New Roman").FontSize(14);  for (int i = 0; i < connectionKlientDetail.Count; i++)  {  doctable.Rows[i + 1].Cells[0].Paragraphs[0].Append(connectionKlientDetail[i].KlientInfo).Font("Times New Roman").FontSize(14);  }  document.InsertParagraph().InsertTableAfterSelf(doctable);  document.Save();  MessageBox.Show("Отчет успешно сформирован!");  Process.Start(pathDocumentDOCX);  break;  case (".xlsx"):  Excel.Application excel;  Excel.Workbook worKbooK;  Excel.Worksheet worKsheeT;  Excel.Range celLrangE;  string pathDocumentXLSX = Session.baseDir + "Список клиентов" + extension;  try  {  excel = new Excel.Application();  excel.Visible = false;  excel.DisplayAlerts = false;  worKbooK = excel.Workbooks.Add(Type.Missing);  worKsheeT = (Microsoft.Office.Interop.Excel.Worksheet)worKbooK.ActiveSheet;  worKsheeT.Name = "Список клиентов";  worKsheeT.Range[worKsheeT.Cells[1, 1], worKsheeT.Cells[1, 8]].Merge();  worKsheeT.Cells[1, 1] = "Список клиентов";  worKsheeT.Cells.Font.Size = 15;  for (int i = 0; i < connectionKlientDetail.Count; i++)  {  worKsheeT.Cells[i + 3, 1] = connectionKlientDetail[i].KlientInfo;  }  worKbooK.SaveAs(pathDocumentXLSX); ;  worKbooK.Close();  excel.Quit();  MessageBox.Show("Отчет успешно сформирован!");  Process.Start(pathDocumentXLSX);  }  catch (Exception ex)  {  MessageBox.Show(ex.Message);  }  finally  {  worKsheeT = null;  celLrangE = null;  worKbooK = null;  }  break;  case (".pdf"):  string pathDocumentPDF = Session.baseDir + "Список клиентов" + extension;  if (File.Exists(Session.baseDir + "Список клиентов.docx"))  {  Word.Application appWord = new Word.Application();  var wordDocument = appWord.Documents.Open(Session.baseDir + "Список клиентов.docx");  wordDocument.ExportAsFixedFormat(pathDocumentPDF, Word.WdExportFormat.wdExportFormatPDF);  MessageBox.Show("Отчет успешно сформирован!");  wordDocument.Close();  Process.Start(pathDocumentPDF);  }  else  MessageBox.Show("Сначала сформируйте отчет .docx");  break;  }  }  catch (Exception)  {  MessageBox.Show("Отсутсвие Ms Office на компьютере. Пожалуйста скачайте его.");  Process.Start("https://www.microsoft.com/ru-ru/microsoft-365/compare-all-microsoft-365-products?tab=1&rtc=1");  }  }  private void ButtonExport\_Click(object sender, RoutedEventArgs e)  {  Thread t = new Thread(new ThreadStart(createExportDoc));  t.Start();  }  private void ComboBoxExport\_SelectionChanged(object sender, SelectionChangedEventArgs e)  {  ComboBoxItem typeItem = (ComboBoxItem)comboBoxExport.SelectedItem;  extension = typeItem.Content.ToString();  } |
| Добавление многопоточности работы с данными в таблицах – Проверка на ввод нужных символов | async Task<bool> Symb(string str)  {  bool znach = false;  await Task.Run(() =>  {  if (  str.Contains("?") || str.Contains("!") || str.Contains("@") ||  str.Contains("#") || str.Contains("№") || str.Contains("~") ||  str.Contains(";") || str.Contains("%") || str.Contains("$") ||  str.Contains("^") || str.Contains("&") || str.Contains(":") ||  str.Contains("\*") || str.Contains("(") || str.Contains(")") ||  str.Contains("\_") || str.Contains("=") || str.Contains("+") ||  str.Contains("/") || str.Contains("|") || str.Contains("[") ||  str.Contains("]") || str.Contains("{") || str.Contains("}") ||  str.Contains("<") || str.Contains(">") || str.Contains("-") ||  str.Contains(",") || str.Contains("`") || str.Contains("."))  znach = true;  });  return znach;  }  private async void BtRegistration\_Click(object sender, RoutedEventArgs e)  {  if (isAdmin)  {  Close();  return;  }  bool znach2 = await Symb(txtBoxSurname.Text);  if (txtBoxSurname.Text == String.Empty || znach2)  {  MessageBox.Show("Есть ошибка в указании Фамилии");  return;  }  bool znach1 = await Symb(txtBoxName.Text);  if (txtBoxName.Text == String.Empty || znach1)  {  MessageBox.Show("Есть ошибка в указании Имени");  return;  }  bool znach3 = await Symb(txtBoxMiddle\_Name.Text);  if (txtBoxMiddle\_Name.Text == String.Empty || znach3)  {  MessageBox.Show("Есть ошибка в указании Отчества");  return;  }  bool znach4 = await Symb(txtBoxTelephone\_Number.Text);  if (txtBoxTelephone\_Number.Text == String.Empty || znach4)  {  MessageBox.Show("Есть ошибка в указании телефонного номера");  return;  }  if (comboBoxID\_Licence.SelectedIndex < 0)  {  MessageBox.Show("Не выбрана категория");  return;  }  if (txtBoxLogin.Text == String.Empty)  {  MessageBox.Show("Не указан логин");  return;  }  if (txtBoxPassword .Text == String.Empty)  {  MessageBox.Show("Не указан пароль");  return;  }  if (txtBoxPassword.Text != txtBoxPassword2.Text)  {  MessageBox.Show("Введенные пароли не совпадают");  return;  }  if (idAuth(txtBoxLogin.Text) >= 0)  {  MessageBox.Show("Такой логин уже есть в базе");  return;  }  if (ch1.IsChecked == false)  {  MessageBox.Show("Проверка на робота не пройдена");  return;  }  } |
| Конфигурация подключения к источнику данных – пользователь выбирает, к какой базе подключаться | Конфигурация:  using Microsoft.Win32;  using System;  using System.Collections.Generic;  using System.Data.SqlClient;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Carscharing  {  class Configuration\_class  {  public static string Empty = "Empty";    public string DS = Empty,// Переменная Data Source  IC = Empty;//Переменная Initial Catalog  public SqlConnection connection = new SqlConnection();    public bool isConnection  {  get  {  return DS != Empty && IC != Empty;  }  }  public void SQL\_Server\_Configuration\_Get()  {  //Создаёт каталог в одном из корней реестра ОС  RegistryKey registry = Registry.CurrentUser;  //Создаёт папку в выбраном коревом каталоге рееста ОС  RegistryKey key = registry.CreateSubKey("Server\_Configuration");  try  {  //Пытаюсь получить значения из переменных в реестре  DS = key.GetValue("DS").ToString();  IC = key.GetValue("IC").ToString();  }  catch  {  DS = Empty;  IC = Empty;  }  finally  {  //Обновление строки подкючения  connection.ConnectionString = "Data Source = " + DS +  "; Initial Catalog = " + IC +  "; Integrated Security = true;";  }  }      }  }  Program.cs  using System;  using System.Collections.Generic;  using System.Linq;  using System.Threading;  using System.Threading.Tasks;  using System.Windows.Forms;  namespace ConfigAppForms  {  static class Program  {  /// <summary>  /// Главная точка входа для приложения.  /// </summary>  ///  public static bool connect = false;  //Определяет есть подключение к БД или нет  private static Mutex \_instanse;  //Класс создания виртуального процесса приложения  private const string \_app\_Name = "WeaponStore\_App";  [STAThread]  static void Main()  {  bool Create\_app;  \_instanse = new Mutex(true, \_app\_Name, out Create\_app);  if (Create\_app)  {  try  {  //В случае если процесс не найден в системе  //создаётся экземпляр класса конфигурации подключения  //к источнику данных  Configuration\_class configuration  = new Configuration\_class();  //Вызывается метод получения информации из реестра  configuration.SQL\_Server\_Configuration\_Get();  //Попытка открыть поддключение к источнику данных  Configuration\_class.connection.Open();  //В случае если подключение открыто  //в переменую присваевается значение истины  connect = true;  }  catch  {  //Стилистика элементов уаправления ОС  Application.EnableVisualStyles();  Application.SetCompatibleTextRenderingDefault(false);  //---НЕ СОЗДАНО--Объявление экземпляра класса  //конфигурации полдключения к источнику данных  Form1 connection = new Form1();  //---НЕ СОЗДАНО---Вывод окна подключения к источнику данных  //В режиме диалогового окна  connection.ShowDialog();  }  finally  {  //Закрытие подключения к источнику данных  Configuration\_class.connection.Close();  //Проверка состояния подключения к источнику данных  switch (connect)  {  //Если пользователь НЕ настроил подключение то  //вывод сообщения об ошибке  case false:  MessageBox.Show("Ошибка подключения " +  "к источнику данных!",  "Bruh",  MessageBoxButtons.OK,  MessageBoxIcon.Error);  //Закрытие приложения  Environment.Exit(0);  break;  //В случе если подключение установлено  case true:  try  {  //Application.EnableVisualStyles();  //Application.SetCompatibleTextRenderingDefault(false);  //Application.Run(new Form2());  }  catch  {  }  break;  }  }  }  Application.EnableVisualStyles();  //Application.SetCompatibleTextRenderingDefault(false);  Application.Run(new Form2());  }  }  }  Form1:  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;  using System.Threading;  namespace ConfigAppForms  {  public partial class Form1 : Form  {  public Form1()  {  InitializeComponent();  }  private void BtChecked\_Click(object sender, EventArgs e)  {  System.Data.SqlClient.SqlConnection sql  = new System.Data.SqlClient.SqlConnection(  string.Format("Data Source = {0}; Initial Catalog =" +  " {1}; Integrated Security = true;", cbServers.Text,  cbDatabases.Text));  try  {  sql.Open();  btConnect.Enabled = true;  }  catch  {  }  finally  {  sql.Close();  }  }  private void Form1\_Load(object sender, EventArgs e)  {  //Вызов класса конфигурации  Configuration\_class configuration  = new Configuration\_class();  //Присвоение event action события  configuration.Server\_Collection  += Configuration\_Server\_Collection; ;  //Обяхвление экземпляра потока  Thread threadServers =  new Thread(configuration.SQL\_Server\_Enumurator);  //Запуск потока  threadServers.Start();  }  private void Configuration\_Server\_Collection(DataTable obj)  {  //Вызов делегата для присвоения в него фрагмента кода  //Через лямбда выражение => в делегат приваиватся код  Action action = () =>  {  //Для каждой строки таблицы в выпадающий список  //Дополнить колекцию пунктов Server Name\Machine Name  foreach (DataRow r in obj.Rows)  {  cbServers.Items.Add(  string.Format(@"{0}\{1}", r[0], r[1]));  }  cbServers.Enabled = true;  btChecked.Enabled = true;  };  //Присвоение фонового потока в основной  Invoke(action);  }  private void BtCancel\_Click(object sender, EventArgs e)  {  Close();  }  private void CbServers\_SelectedIndexChanged(object sender, EventArgs e)  {  Configuration\_class configuration  = new Configuration\_class();  configuration.ds = cbServers.SelectedItem.ToString();  configuration.Conection\_Checked  += Configuration\_Conection\_Checked;  Thread thread  = new Thread(configuration.SQL\_Data\_Base\_Checking);  thread.Start();  }  private void Configuration\_Conection\_Checked(bool obj)  {  switch (obj)  {  //Если подключение выполнено верно то появляется сообщение  case true:  MessageBox.Show("Проверка выполнена!");  Action action = () =>  {  //Повторение метода выбора  Configuration\_class configuration\_coll  = new Configuration\_class();  configuration\_coll.Data\_Base\_Collection  += Configuration\_Data\_Base\_Collection;  Thread threadBases  = new Thread(configuration\_coll.SQL\_Data\_Base\_Collection);  threadBases.Start();  btConnect.Enabled = true;  };  Invoke(action);  break;  case false:  //Вслучае если нет подключения повторяем сбор данных  //о сервере  Configuration\_class configuration  = new Configuration\_class();  configuration.Server\_Collection  += Configuration\_Server\_Collection;  Thread threadServers  = new Thread(configuration.SQL\_Server\_Enumurator);  threadServers.Start();  break;  }  }  private void Configuration\_Data\_Base\_Collection(DataTable obj)  {  Action action = () =>  {  cbDatabases.Items.Clear();  foreach (DataRow r in obj.Rows)  {  cbDatabases.Items.Add(r[0]);  }  cbDatabases.Enabled = true;  btChecked.Enabled = true;  };  Invoke(action);  }  private void BtConnect\_Click(object sender, EventArgs e)  {  switch (cbDatabases.Text == "")  {  case true:  //В случае если поле не заполнено  MessageBox.Show("Не выбрана нужная база данных!",  "Каршеринг",  MessageBoxButtons.OK, MessageBoxIcon.Warning);  cbDatabases.Focus();  break;  case false:  Configuration\_class configuration  = new Configuration\_class();  //Присвоение конфигурации в реестр ОС  configuration.SQL\_Server\_Configuration\_Set(  cbServers.Text, cbDatabases.Text);  //Присвоение в переменную значение  //о правильности подключения  Program.connect = true;  //Перезапуск программы  Application.Restart();  break;  }  }  }  }  Form2:  using Carscharing;  using System;  using System.Collections.Generic;  using System.ComponentModel;  using System.Data;  using System.Diagnostics;  using System.Drawing;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.Windows.Forms;  using System.Windows.Forms.Integration;  namespace ConfigAppForms  {  public partial class Form2 : Form  {  public Form2()  {  InitializeComponent();      }  private void Form1\_Load(object sender, EventArgs e)  {  }  private void Button1\_Click(object sender, EventArgs e)  {  Process.Start(@"C:\Users\1\Desktop\Новый семестр-новые страдания\Курсовой\WeaponStore\bin\Debug\WeaponStore.exe");  Close();  }  }  } |
| REAL TIME – При добавлении записи из БД, она мгновенно появляется в таблице приложения | БД:  Alter database [Weapon\_Store] set enable\_broker with rollback immediate  go  Visual:  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  using System.Windows;  using System.Windows.Controls;  using System.Windows.Data;  using System.Windows.Documents;  using System.Windows.Input;  using System.Windows.Media;  using System.Windows.Media.Imaging;  using System.Windows.Shapes;  using System.Data;  using System.Data.SqlClient;  using WeaponStore.TableConnection;  namespace WeaponStore  {  /// <summary>  /// Логика взаимодействия для Ammo.xaml  /// </summary>  public partial class Ammo : Window  {  private TableConnection.ConnectionAmmo connectionAmmo;  List<TableConnection.ConnectionAmmo> Ammos;  public string extension = string.Empty;  private string QR = "";  public Ammo()  {  InitializeComponent();  }  private void Window\_Loaded(object sender, RoutedEventArgs e)  {  dgAmmo.Visibility = Visibility.Hidden;  Ammos = (new DBProcedures()).getAmmoList();  dgAmmo.ItemsSource = Ammos;  dgAmmo.Columns[0].Visibility = Visibility.Collapsed;  Action action = () =>  {  Ammos = (new DBProcedures()).getAmmoList();  dgAmmo.ItemsSource = Ammos;  };  Dispatcher.Invoke(action);  }  } |

## **3.2. Оценка качества программного продукта**

### **3.2.1. Модель оценивания**

Таблица 3.2 - модель оценивания ПП

| Фактор | Метрика | Критерии | Ранжирование (0-5) |
| --- | --- | --- | --- |
| Функциональная пригодность | Точность | Все данные выводятся на русском языке в понятном пользователю виде. | 5 |
| Все данные выводятся поровну на английском и русском языках в понятном пользователю виде. | 4 |
| Все данные выводятся на английском языке в понятном пользователю виде. | 3 |
| Данные выводятся без редактирования, напрямую из БД. | 2 |
| Данные выводятся, но не правильно | 1 |
| Данные не выводятся | 0 |
| Защищённость | Данные защищаются с помощью уникальной аунтефикации и с помощью средств защиты данных (маски и т.п.) | 5 |
| Данные защищаются с помощью уникальной аунтефикации | 4 |
| Данные защищаются с помощью «Секретного слова» с помощью средств защиты данных. | 3 |
| Данные защищаются с помощью средств аутентификации | 2 |
| Данные защищаются с помощью «Секретного слова» без средств защиты данных. | 1 |
| Данные не защищаются | 0 |
| Способность к взаимодействию | Взаимодействует абсолютно со всем ПО. | 5 |
| Взаимодействует с сторонним ПО от официальных разработчиков | 4 |
| Взаимодействует с сторонним ПО, заранее установленное в опции к данному программному продукту | 3 |
| Взаимодействует с подключением стороннего ПО (только с внедрением от разработчика) | 2 |
| Взаимодействует только с Sql Server | 1 |
| Не с чем не взаимодействует | 0 |
| Согласованность | Полное согласование с заказчиком, отчет от программиста на каждом этапе выполнения работы | 5 |
| Согласование с заказчиком в виде документации и итоговым заверением заказчиком | 4 |
| Проект обговаривается с заказчиком лично с возможность вносить поправки в некоторые пункты, которые были согласованы ранее | 3 |
| Проект обговаривается с заказчиком лично и дистанционно | 2 |
| Проект обговаривается с заказчиком дистанционно | 1 |
| При попытке обговорить проект, заказчик уходил от разговора или не желал обговаривать проект. | 0 |
| Надежность | Стабильность | Стабильность при большинстве процессов | 5 |
| Стабильность с редкими зависаниями | 4 |
| Частые зависания системы | 3 |
| Частые зависания почти не возможное выполнения процессов | 2 |
| Невозможность выполнения процессов | 1 |
| При любых процессах программа работает с одинаковой скоростью | 0 |
| Надежность  Стабильность | Стабильность (частота отказов) | Программный продукт уведомляет пользователя о наличии какой либо ошибки в эксплуатации данного ПП | 5 |
| Программный продукт имеет защищенность от ввода данных неверного формата, но не уведомляют о них | 4 |
| Программа имеет защищенность, но не распространяется на все виды ошибок | 3 |
| Программа имеет защищенность, но работает не корректно | 2 |
| Программный продукт не имеет ограничений по вводу на все | 1 |
| Программный продукт отказывает при наличии какой либо ошибки | 0 |
| Устойчивость к ошибкам | Программный продукт полностью устойчив к ошибкам, уведомляет о каждом не верно проделанном действии пользователя | 5 |
| Программный продукт устойчив к ошибкам, но не уведомляет о них | 4 |
| ПП устойчив к основному количеству ошибок | 3 |
| Малая часть ПП устойчива к ошибкам | 2 |
| Программный продукт выдает ошибку | 1 |
| ПП полностью не устойчив к ошибкам | 0 |
| Восстанавливаемость | Восстанавливаемость происходит сразу же после ошибки, о которой пользователь был уведомлен и согласился с ошибкой | 5 |
| Восстанавливаемость происходит сама при какой либо ошибки | 4 |
| Восстанавливаемость происходит после уведомления пользователя об ошибки с последующим перезапуском ПП | 3 |
| Восстанавливаемость происходит после автоматического перезапуска ПП | 2 |
| Восстанавливаемость происходит только после перезапуска ПП, сделать это может сам пользователь | 1 |
| Восстанавливаемость происходит только после перезапуска ПП и сделать это может только администратор БД | 0 |
| Применяемость | Понятность | Программный продукт имеет полностью понятный код с соблюдением правил табуляции и с наличием комментариев к каждому пункту | 5 |
| Программный продукт имеет понятный код с комментариями | 4 |
| Программный продукт имеет код с комментариями | 3 |
| Программный продукт обладает не понятным кодом без комментариев | 2 |
| ПП не обладает никакими правилами табуляции, полное отсутствие комментариев | 1 |
| ПП обладает слитным кодом который является абсолютно непонятным оператору | 0 |
| Обучаемость | Пользователь должен обладать базовыми знаниями о владении ПК и о специфики работы с ПП | 5 |
| Пользователь должен обладать какими либо базовыми знаниями в сфере ПП | 4 |
| Пользователь должен ознакомиться с правилами пользования данного ПП. | 3 |
| Пользователь должен пройду курс по обучению пользования ПП | 2 |
| Пользователь должен обладать определенными знаниями в сфере разработки и пользования ПП, ознакомиться с требованиями пользования данным ПП | 1 |
| Программа сложна в использовании | 0 |
| Простота использования | Данный программный продукт легок в использовании, обладает понятным механизмом работы и взаимодействия с ним | 5 |
| Данный ПП обладает понятным механизмом пользования | 4 |
| Данный ПП обладает достаточно сложным механизмом. | 3 |
| Данный ПП обладает достаточно сложным механизмом и не понятным интерфейсом | 2 |
| Данный ПП обладает сложным механизмом и не понятным интерфейсом | 1 |
| Данный ПП обладает сложным механизмом с не понятным интерфейсом, требуется ознакомление с работой в данном ПП | 0 |
| Эффективность  Фактор | Временная экономичность | Программный продукт обладает моментальным откликом после совершения каких либо действий при условий нахождения сервера на машине | 5 |
| Программный продукт обладает быстрым откликом после совершения каких либо действий в некоторых секторах ПП | 4 |
| Программный продукт обладает долгим откликом после совершения каких либо действий в не которых секторах ПП | 3 |
| Программный продукт обладает долгим откликом после совершения каких либо действий во всех секторах ПП | 2 |
| Программный продукт имеет очень долгий отклик | 1 |
| Программный продукт имеет очень длинный отклик | 0 |
| Ресурсная экономичность | Программный продукт не требует высоких требований к характеристикам ПК, но на слишком старом оборудовании могут происходить сбои | 5 |
| Программный продукт обладает минимальными требованиями по запуску данного ПП | 4 |
| Программный продукт требует как минимум средних характеристик пк | 3 |
| Программный продукт требует высоких характеристик ПК | 2 |
| Программный продукт с трудом запускается на любой машине | 1 |
| Не запускается на слабых пк, дает сбои на мощных | 0 |
| Сопровождаемость  Тест | Удобство для анализа | Программный продукт при наличии ошибок, выведет сообщение в среду с типом ошибки и ее описанием в котором она будет обозначена, это делает процесс исправления ошибок легче | 5 |
| Программный продукт при наличии ошибок, выведет сообщение в среду с типом и описанием ошибки, однако некорректно ее описывает | 4 |
| Программный продукт выводит ошибку на экран но не в среду разработки | 3 |
| Программный продукт выводит ошибку на экран и в среду, но требуется долгое время для ее анализа | 2 |
| Программный продукт выводит ошибки на экране и в среду с нераскрытым смыслом самой ошибки | 1 |
| Программный продукт не выводит и не сообщает об ошибках | 0 |
| Изменяемость | Программный код имеет комментарии и соблюдены все правила табуляции и у всех компонентов их названия соответствуют действиям, что позволяет программисту (стороннему в том числе) легче ориентироваться в коде | 5 |
| Программный код имеет комментарии и у всех компонентов их названия соответствуют действиям, что позволяет программисту (стороннему в том числе) легче ориентироваться в коде | 4 |
| Программный код имеет комментарии и все переменные и действия обозначены правильно | 3 |
| Программный код не имеет комментарии и все переменные и действия обозначены правильно | 2 |
| Программный код не имеет комментарии и все переменные и действия обозначены не правильно | 1 |
| Программный код не имеет комментариев, название переменных не соответствует их действиям, отсутствует правила табуляции | 0 |
| Устойчивость | В ПП присутствуют таблицы, которые на прямую или частично зависят друг от друга и при отказе работы одной из таблиц, другая продолжает функционировать | 5 |
| В ПП присутствуют связанные таблицы, однако при отказе одной из таблиц другая функционирует не стабильно | 4 |
| В ПП разные таблицы при отказе одной из них продолжают функционировать, а какие-то нет | 3 |
| В ПП при отказе одной из зависимых таблиц одна из них полностью перестает функционировать | 2 |
| В ПП при отказе одной из зависимых таблиц полностью перестают функционировать две зависимые таблицы | 1 |
| Программа не имеет устойчивости к ошибкам в функционировании таблиц | 0 |
| Тестируемость | Данный программный продукт легок в тестировании, т.к. обладает понятным, полностью прокомментированным кодом в котором присутствует табуляция и грамотное обозначение компонентов, а так же удобный интерфейс | 5 |
| Данный программный продукт обладает средней сложности тестирования, т.к. обладает понятным, прокомментированным кодом в котором присутствует табуляция, но компоненты имеют непонятные названия | 4 |
| Данный программный продукт сложно протестировать, т.к. код не закомментирован, присутствует табуляция, компоненты имеют непонятные названия | 3 |
| Данный программный продукт сложно протестировать, т.к. код не закомментирован, отсутствует табуляция, компоненты имеют непонятные названия | 2 |
| Данный программный продукт практически не подлежит тестированию, т.к. код написан сплошным текстом, не закомментирован, отсутствует какая-либо табуляция. | 1 |
| Данный программный продукт не подлежит тестированию. | 0 |
| Мобильность | Адаптация | Данный программный продукт работает на любой ОС (начиная с минимальной, которую установил разработчик) наличие прикладных программ не обязательно (Microsoft SQL Server и Microsoft Visual Studio (Net Framework) т.к. все работает от сервера | 5 |
| Данный программный продукт работает на любой ОС (начиная с минимальной, которую установил разработчик) наличие прикладных программ обязательно (Microsoft SQL Server и Microsoft Visual Studio (Net Framework) | 4 |
| Данный программный продукт работает на ОС (выше Windows XP) наличие прикладных программ обязательно (Microsoft SQL Server и Microsoft Visual Studio (Net Framework) | 3 |
| Данный программный продукт работает на ОС (выше Windows Vista) наличие прикладных программ обязательно (Microsoft SQL Server 2018 и Microsoft Visual Studio (Net Framework 2018) | 2 |
| Данный программный продукт работает только на той ОС, под которую был разработан данный ПП, наличие прикладных программ обязательно (Microsoft SQL Server 2018 и Microsoft Visual Studio (Net Framework 2018) | 1 |
| Данный программный продукт требует наличие той системы, на которой и создавался данной ПП. | 0 |
| Простота внедрения | Данный программный продукт легко внедряем, т.к. он совмещен с разными ОС | 5 |
| Данный программный продукт легко внедряем, т.к. он совмещен с разными ОС, единственное исключение, требуется наличие Framework | 4 |
| Данный программный продукт трудно внедряем, т.к. он разработан под определенные платформы ОС (Windows), требуется наличие Framework | 3 |
| Данный программный продукт трудно внедряем, т.к. он разработан под определенные платформы и версии ОС (начиная с Windows XP), требуется наличие Framework | 2 |
| Данный программный продукт трудно внедряем, т.к. он разработан под определенные платформы и версии ОС (начиная с Windows 7), требуется наличие Framework | 3 |
| Данный программный продукт трудно внедряем, т.к. он разработан под одну платформу и версию ОС, требуется наличие Framework | 0 |
| Соответствие | Данный программный продукт полностью соответствует требованию заказчика, стандартам разработки и различным ГОСТам | 5 |
| Данный программный продукт полностью соответствует требованию заказчика, присутствуют минимальные отклонения от стандартов разработки | 4 |
| Данный программный продукт не полностью соответствует требованию заказчика и стандартам разработки | 3 |
| Данный программный продукт не полностью соответствует требованию заказчика, существенно отклоняется от правил разработки | 2 |
| Данный программный продукт не соответствует требованию заказчика, существенно отклоняется от правил разработки | 1 |
| Данный программный продукт полностью не соответствует требованию заказчика, отклоняется от правил разработки | 0 |

* Итоги оценивания

1. Точность

Данный критерий был оценён на 5 балла, так как все данные выводятся в виде на русском языке понятном пользователю.

1. Защищённость

Данный критерий был оценен на 5, так как данные защищаются с помощью уникальной аутентификации с средствами защиты информации.

1. Согласованность

Данный критерий был оценен на 0, так как при попытке обговорить какие-либо его аспекты с заказчиком не были проведены по причине нежелания заказчика.

1. Стабильность

Данный критерий был оценен на 5, так как ПП стабилен при большинстве процессов.

1. Способность к взаимодействию

Данный критерий был оценен на 4, так как взаимодействует с сторонним ПО от официальных разработчиков (MS Word и Excel).

1. Стабильность (частота отказов)

Данный критерий был оценен на 3, так как программа имеет защищенность, но не распространяется на все виды ошибок.

1. Устойчивость к ошибкам

Данный критерий был оценен на 3, так как программный продукт устойчив к большинству ошибок, но уведомляет о каждом не верно проделанном действии пользователя.

1. Восстанавливаемость

Данный критерий был оценен на 5, так как восстанавливаемость происходит сразу же после ошибки, о которой пользователь был уведомлен и согласился с ошибкой.

1. Понятность

Данный критерий был оценен на 2, так как программный продукт обладает не понятным кодом без комментариев.

1. Обучаемость

Данный критерий был оценен на 5, так как пользователь должен обладать базовыми знаниями о владении ПК и о специфики работы с ПП.

1. Простота использования

Данный критерий был оценен на 5, так как данный ПП обладает понятным механизмом пользования.

1. Временная экономичность

Данный критерий был оценен на 4, так как программный продукт обладает быстрым откликом после совершения каких-либо действий в некоторых секторах ПП.

1. Ресурсная экономичность

Данный критерий был оценен на 4, так как программный продукт обладает минимальными требованиями по запуску данного ПП.

1. Удобство для анализа

Данный критерий был оценен на 4, так как программный продукт при наличии ошибок, выведет сообщение в среду с типом и описанием ошибки, однако некорректно ее описывает.

1. Изменяемость

Данный критерий был оценен на 2, так как программный код не имеет комментарии и все переменные и действия обозначены правильно.

1. Устойчивость

Данный критерий был оценен на 4, так как в ПП присутствуют связанные таблицы, однако при отказе одной из таблиц другая функционирует не стабильно.

1. Тестируемость

Данный критерий был оценен на 5, программный продукт легко протестировать с помощью метода ручного тестирования, даже при учете не прокомментированного кода.

1. Адаптация

Данный критерий был оценен на 5, так как программный продукт работает на любой ОС (начиная с минимальной, которую установил разработчик) наличие прикладных программ не обязательно (Microsoft SQL Server и Microsoft Visual Studio (Net Framework) т.к. все работает от сервера.

1. Простота внедрения

Данный критерий был оценен на 3, так как программный продукт трудно внедряем, т.к. он разработан под определенные платформы и версии ОС (начиная с Windows 7), требуется наличие Framework.

1. Соответствие

Данный критерий был оценен на 3, так как программный продукт не полностью соответствует требованию заказчика и стандартам разработки.

Исходя из оценки программного продукта, можно сделать вывод, что программный продукт прост в использовании, однако отсутствие внешнего контроля со стороны заказчика и сжатые сроки привели к тому, что некоторые функции работают топорно и требую доработки, но в основном ПП исправно выполняет свои функции.

## **3.3. Итоги разработки**

* Хронология разработки

Таблица 3.7. Хронология разработки информационной системы

|  |  |  |
| --- | --- | --- |
| Версия | Изменение | Дата изменения |
| 1.1.1 | Создание Базы данных и таблиц | 18.05.2020 |
| 1.1.2 | Настройка моделей и установка межтабличных связей на уровне приложения | 18.05.2020 |
| 1.1.3 | Создание представлений | 18.05.2020 |
| 1.1.4 | Создание интерфейса | 20.05.2020 |
| 1.1.5 | Выгрузка данных из БД в табличные и другие элементы управления | 25.05.2020 |
| 1.1.6 | Добавление в приложение функций добавления, изменения и удаления | 29.05.2020 |
| 1.1.7 | Добавление функций авторизации и регистрации | 04.06.2020 |
| 1.1.9 | Добавление справочников | 08.06.2020 |
| 1.2.0 | Добавление списка клиентов | 08.06.2020 |
| 1.2.1 | Добавление каталога | 10.06.2020 |
| 1.2.2 | Добавление корзины | 15.06.2020 |
| 1.2.3 | Добавление экспорта | 17.06.2020 |
| 1.2.4 | Шлифовка интерфейса | 20.06.2020 |

* Выходная версия.

Максимальная оценка программы 100.

В результате оценки программный продукт набрал 76.

Таблица 3.3 Критерии определения выходной версии информационной системы

|  |  |
| --- | --- |
| Выходная версия | Балл |
| RP – поставка (Окончательная поставка) | 90-100 |
| PB – поставка (Поставка бета - версии) | 70-89 |
| PA – поставка (Поставка альфа - версии) | 50-69 |
| ES – поставка (Поставка прототипа) | 0-49 |

Данный программный продукт оценён на 76 баллов соответственно - PB – поставка (Поставка бета - версии).

# **ЗАКЛЮЧЕНИЕ**

В рамках курсового проекта МДК 02.01 «Разработка, внедрение и адаптация программного обеспечения отраслевой направленности» было проведено изучение предметной области «Магазин оружия», проанализированы подсистемы: «Отдел кадров», «Отдел снабжения» и «Отдел продаж».

Был проведен анализ предметной области по методологиям:

* IDEF0 и DFD (Process Modeler R7);
* EPC (Aris Express)

Далее была спроектирована ИС, созданы даталогическая и инфологическая модели с помощью средства ERwin, Process Modeler R7 и созданы алгоритмы для функций.

Была реализована база данных с помощью языка SQL, а именно: таблицы, представления, процедуры, функции.

В итоге было разработано WPF приложение, выполняющее все функции, заявленные в пункте Цели разработки.

WPF приложение требует не значительных доработок по коду (комментариев) и реализации функций разгрузки поставок и доработки корзины.

К WPF приложению можно отнести следующее:

* Простой и понятный интерфейс;
* Удобная регистрация (с подсказками) и авторизация;
* Возможность предоставить клиенту выбрать интересующий его товар;
* Автоматизация учета товаров, регистрации пользователей разных категорий (сотрудники и клиенты);

# **СПИСОК ИСПОЛЬЗУЕМЫХ МАТЕРИАЛОВ**

1. Прошлый ПП на WPF с 1 семестра 3 курса;
2. Отчеты по РВИАПООН с 1 семестра 3 курса;
3. Алгоритм шифрования RSA <https://studwood.ru/1685074/informatika/preimuschestva_nedostatki_algoritma_shifrovaniya>
4. Форумы для программистов CyberForum.
5. Лабораторная работа № 6 «Проектирование структуры базы данных. Создание ER – моделей.»
6. Практическая работа № 3 «Разработка простых форм на базе технологии WPF»
7. Практическая работа № 5 «Разработка вывода данных из таблиц БД в компонент DataGrid»
8. Практическая работа № 7 «Реализация функций добавления, изменения, удаления данных в WPF»
9. Практическая работа № 11 «Реализация системы аутентификации WPF»
10. Практическая работа № 22 «Определение метрик для реализованного программного продукта.»
11. Практическая работа № 24 «Подготовка модели оценивания программного продукта.»
12. Практическая работа № 12 «Разработка сценариев и тестов работы приложения.»
13. Практическая работа № 13 «Проведение ручного тестирования.»
14. Средство реализации SSMS <https://info-comp.ru/softprodobes/526-environment-sql-server-management-studio.html>
15. Средство реализации Visual Studio <https://win10soft.ru/169-visual-studio.html>