

УТВЕРЖДАЮ

Руководитель практики от Университета

Преподаватель, Шахторин И.Д.

*(должность, ФИО)*

МП \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*(Подпись)*

Дата «29» марта 2022 год

**ОТЧЕТ**

**ПО УЧЕБНОЙ ПРАКТИКЕ**

ПМ.03 Ревьюирование программных продуктов

|  |  |
| --- | --- |
| Костров Илья Александрович | |
| *(Ф.И.О. обучающегося)* | |
| 09.02.07 Информационные системы и программирование | |
| *(специальность)* | |
|  | |
| Учебная группа | ИСПк-403-52-00 |
|  |  |
| Место прохождения практики | ФГБОУ ВО «Вятский государственный университет», |
| Колледж ВятГУ | *(наименование организации, структурного подразделения организации)* |
|  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Итоговая оценка: |  | | | | |
| Руководитель  практики от университета |  |  |  |  | Шахторин И.Д. |
|  | *(дата)* |  | *(подпись)* |  | *(Ф.И.О.)* |

Киров, 2022 г.



ИНДИВИДУАЛЬНОЕ ЗАДАНИЕ НА ПРАКТИКУ

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Ф.И.О. обучающегося | | | Костров Илья Александрович | | | | | | |
| Специальность | | | 09.02.07 Информационные системы и программирование | | | | | | |
| Учебная группа | | | ИСПк-403-52-00 | | | | | | |
| Вид практики | | | учебная практика | | | | | | |
| Сроки прохождения практики с | | | | 23.03.2022 | по | 29.03.2022 | | | |
| Место прохождения практики | | | | ФГБОУ ВО «Вятский государственный университет», Колледж ВятГУ | | | | | |
|  | | (наименование организации, структурного подразделения организации) | | | | | | |
| № | Виды работ, выполняемых обучающимися во время практики | | | | | | Объем работ (час) | Формируемые компетенции | | |
| 1 | Пройти инструктаж по ознакомлению с правилами внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологическими правилами и гигиеническими нормативами, а также вводный инструктаж и инструктаж на рабочем месте.  Изучить проектную документацию. Создать диаграммы программного продукта. | | | | | | 4 | ПК3.5, ОК02, ОК03 | | |
| 2 | Настроить рабочее окружение. Настроить систему контроля версий. | | | | | | 2 | ПК3.4, ОК03, ОК06, ОК07, ОК09 | | |
| 3 | Решить набор задач по созданию меню клиента и присоединению БД. | | | | | | 20 | ПК3.3, ПК3.4, ОК01, ОК02, ОК04, ОК08, ОК09 | | |
| 4 | Провести тестирование и отладку полученного кода. | | | | | | 4 | ПК3.2, ПК3.3, ОК02, ОК09 | | |
| 5 | Подготовить отчетную документацию, включая детальное описание разработанных и использованных алгоритмов, провести ревьюирование кода программного продукта. | | | | | | 4 | ПК3.1, ПК3.2 ОК09, ОК10 | | |
| 6 | Подготовиться к сдаче промежуточной аттестации | | | | | | 2 | ОК03, ОК04, ОК05 | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Индивидуальное задание на практику разработано в соответствии с рабочей программой практики. | | | | | |
| Руководитель  практики от университета | 23.03.2022 |  |  |  | Шахторин И.Д. |
|  | (дата) |  | (подпись) |  | (Ф.И.О.) |

|  |  |
| --- | --- |
| С индивидуальным заданием ознакомлен(а) | 23.03.2022 |
|  | (дата, подпись обучающегося) |

**ОТЗЫВ РУКОВОДИТЕЛЯ ПРАКТИКИ**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Ф.И.О. обучающегося | | Костров Илья Александрович | | | |
| Специальность | | 09.02.07 Информационные системы и программирование | | | |
| Учебная группа | | ИСПк-403-52-00 | | | |
| Вид практики | | учебная практика | | | |
| Сроки прохождения практики с | | | 23.03.2022 | по | 29.03.2022 |
| Место прохождения практики | | | ФГБОУ ВО «Вятский государственный университет», Колледж ВятГУ | | |
|  | (наименование организации, структурного подразделения организации) | | | | |

ВИДЫ И КАЧЕСТВО ВЫПОЛНЕНИЯ РАБОТ

|  |  |  |  |
| --- | --- | --- | --- |
| Вид работ | Критерий выполнения работ | | |
| Выполнены полностью самостоятельно | Выполнены с незначительной помощью наставника | Выполнены с помощью наставника |
| Настроить рабочее окружение. Настроить систему контроля версий. |  |  |  |
| Решить набор задач по созданию меню клиента и присоединению БД. |  |  |  |
| Провести тестирование и отладку полученного кода. |  |  |  |
| Подготовить отчетную документацию, включая детальное описание разработанных и использованных алгоритмов, провести ревьюирование кода программного продукта. |  |  |  |
| Подготовиться к сдаче промежуточной аттестации |  |  |  |

Обучающийся ознакомлен с правилами внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологическими правилами и гигиеническими нормативами, а также прошел вводный инструктаж и инструктаж на рабочем месте.

Во время прохождения учебной практики обучающимся освоены следующие профессиональные и общие компетенции:

|  |  |  |  |
| --- | --- | --- | --- |
| Наименование компетенции | Показатели оценки | Оценка | |
| Освоена | Не освоена |
| ПК 3.1. Осуществлять ревьюирование программного кода в соответствии с технической документацией. | Способен анализировать корректность через просмотр программного кода на предмет соответствия предъявляемым требованиям. | V |  |
| ПК 3.2. Выполнять измерение характеристик компонент программного продукта для определения соответствия заданным критериям. | Способен выполнить измерение характеристик компонент программного продукта в целях выявления необходимых критериев и их подтверждения. | V |  |
| ПК 3.3. Производить исследование созданного программного кода с использованием специализированных программных средств с целью выявления ошибок и отклонения от алгоритма. | Способен проводить экспертизу программного кода в целях устранения потенциально некорректных решений и достижения требуемых показателей работоспособности. | V |  |
| ПК 3.4. Проводить сравнительный анализ программных продуктов и средств разработки, с целью выявления наилучшего решения согласно критериям, определенным техническим заданием. | Способен сравнивать и анализировать программное обеспечение в целях подбора эффективного средства разработки для решения поставленных задач. | V |  |
| ПК 3.5. Проводить исследование проектной документации программного продукта. | Способен оформлять документацию, включающую в себя описание программного продукта на доступном и понятном уровне. | V |  |
| ОК 01. Выбирать способы решения задач профессиональной деятельности, применительно к различным контекстам. | Способен анализировать специализированную информацию и находить оптимальные пути решения задач профессиональной деятельности. | V |  |
| ОК 02. Осуществлять поиск, анализ и интерпретацию информации, необходимой для выполнения задач профессиональной деятельности. | Способен анализировать специализированную информацию и находить оптимальные пути решения задач профессиональной деятельности. | V |  |
| ОК 03. Планировать и реализовывать собственное профессиональное и личностное развитие. | Способен находить пути улучшения имеющихся решений, позволяющих повысить их общий качественный уровень. | V |  |
| ОК 04. Работать в коллективе и команде, эффективно взаимодействовать с коллегами, руководством, клиентами. | Способен конструктивно обмениваться информацией с коллегами, грамотно формулировать запросы в целях получения разъясняющей информации. | V |  |
| ОК 05. Осуществлять устную и письменную коммуникацию на государственном языке с учетом особенностей социального и культурного контекста. | Способен конструктивно обмениваться информацией с коллегами, грамотно формулировать запросы в целях получения разъясняющей информации. | V |  |
| ОК 06. Проявлять гражданско-патриотическую позицию, демонстрировать осознанное поведение на основе традиционных общечеловеческих ценностей. | Способен конструктивно обмениваться информацией с коллегами, грамотно формулировать запросы в целях получения разъясняющей информации. | V |  |
| ОК 07. Содействовать сохранению окружающей среды, ресурсосбережению, эффективно действовать в чрезвычайных ситуациях. | Способен эффективно использовать компьютерное время, а также материальные ресурсы, необходимые для решения поставленных задач. | V |  |
| ОК 08. Использовать средства физической культуры для сохранения и укрепления здоровья в процессе профессиональной деятельности и поддержания необходимого уровня физической подготовленности. | Способен соблюдать требования внутреннего трудового распорядка организации, охраны труда и техники безопасности в целях сохранения собственного здоровья. | V |  |
| ОК 09. Использовать информационные технологии в профессиональной деятельности. | Способен применять стек современных средств разработки ПО для решения задач профессиональной деятельности в заданном контексте. | V |  |
| ОК 10. Пользоваться профессиональной документацией на государственном и иностранном языке. | Способен использовать в своей работе специализированную документацию. | V |  |

**Краткая характеристика работы обучающегося**

|  |
| --- |
| Программа практики выполнена в полном объеме. Все виды работ выполнялись в срок, без |
| существенных замечаний. В достаточной степени была проявлена самостоятельность и |
| умение грамотно пользоваться источниками информации, находящимися в свободном |
| доступе. |

|  |  |  |
| --- | --- | --- |
|  |  | Руководитель практики  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ /Шахторин И.Д.  Подпись ФИО  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_преподаватель\_\_\_\_\_\_\_\_\_\_\_\_  (должность)  Дата «\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 202\_\_ год |

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 7](#_Toc99354701)

[1. СВЕДЕНИЯ О РАБОТЕ, ВЫПОЛНЕННОЙ В ПЕРИОД ПРОХОЖДЕНИЯ УЧЕБНОЙ ПРАКТИКИ 8](#_Toc99354702)

[2. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ 9](#_Toc99354703)

[3. ПРОЕКТИРОВАНИЕ ЛОГИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ 10](#_Toc99354704)

[4. РАЗРАБОТКА ФИЗИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ 13](#_Toc99354705)

[5. РАЗРАБОТКА ТЕХНИЧЕСКИХ ТРЕБОВАНИЙ К СЕРВЕРУ БАЗ ДАННЫХ. КОНФИГУРИРОВАНИЕ СЕТИ 16](#_Toc99354706)

[6. РАЗРАБОТКА БАЗЫ ДАННЫХ И ТАБЛИЦ, ДОБАВЛЕНИЕ, УДАЛЕНИЕ ДАННЫХ 19](#_Toc99354707)

[7. ОПИСАНИЕ МЕТОДОВ РЕВЬЮВИРОВАНИЯ 20](#_Toc99354708)

[ЗАКЛЮЧЕНИЕ 21](#_Toc99354709)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 22](#_Toc99354710)

ПРИЛОЖЕНИЕ А [ЛИСТИНГ ПРОГРАММЫ 24](#_Toc99354712)

[ПРИЛОЖЕНИЕ Б.](#_Toc99354713) [ФОРМЫ ПРОЕКТА 28](#_Toc99354714)

[ПРИЛОЖЕНИЕ В. ТЕХНИЧЕСКОЕ ЗАДАНИЕ 31](#_Toc99354715)

[ТЕХНИЧЕСКОЕ ЗАДАНИЕ 31](#_Toc99354716)

**ВВЕДЕНИЕ**

Учебная практика ПМ.03 Ревьюирование программных продуктов проходила на базе Колледжа ФГБОУ ВО «Вятский государственный университет» в период с 23.03.2022 по 29.03.2022.

Цель практики: комплексное освоение обучающимися вида профессиональной деятельности «Ревьюирование программных продуктов», формирование общих и профессиональных компетенций, приобретение опыта практической работы по специальности.

Задачи практики:

1. Описать предметную область.
2. Спроектировать логическую модель базы данных.
3. Разработать физическую модель базы данных.
4. Разработать технические требования к серверу баз данных. Выполнить конфигурирование сети.
5. Установить и настроить сервер баз данных.
6. Разработать базу данных и таблиц, добавить, удалить данные.
7. Описать использованные методы ревьюирования.

# **СВЕДЕНИЯ О РАБОТЕ, ВЫПОЛНЕННОЙ В ПЕРИОД ПРОХОЖДЕНИЯ УЧЕБНОЙ ПРАКТИКИ**

В период с 23.03.2022 по 29.03.2022 при прохождении учебной практики ПМ.03 на базе ФГБОУ ВО «Вятский государственный университет» был выполнен следующий перечень работ, представленный в таблице 1.

Таблица 1 – Сведения о работе, выполненной в период практики

| Дата | Краткое содержание выполненных работ |
| --- | --- |
| 23.03.2022 | Прохождение инструктажей по ознакомлению с правилами внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологическими правилами и гигиеническими нормативами, а также вводный инструктаж и инструктаж на рабочем месте. |
| 23.03.2022 | Изучение проектной документации. Создание диаграмм для программного продукта. |
| 23.03.2022 | Настройка рабочего окружения. Настройка системы контроля версий. |
| 24.03.2022-27.03.2022 | Решение набора задач по созданию меню клиента и присоединению БД. |
| 28.03.2022 | Тестирование и отладка полученного кода |
| 28.03.2022 | Подготовка отчетной документации, включая детальное описание разработанных и использованных алгоритмов, ревьюирования кода программного продукта. |
| 29.03.2022 | Подготовка к сдаче и сдача промежуточной аттестации |

23.03.2022 (подпись)

# **ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ**

Компьютерный магазин предоставляет клиентам возможность локально приобрести товары для своей электроники без необходимости ожидания доставки, так как оперативность и удобность в подборе комплектующих пользователю выходит на первый уровень.

Надежная и эффективная работа программы по учету и автоматизации бизнес-процессов магазина позволяет уменьшить количество времени для владельца.

Объектом автоматизации является информационная система по учету деятельности магазина. База данных хранит в себе данные о: персонале, товарной группе, имеющихся товарах, пользователях и заказах. При оформлении заказа, оператор вносит данные о пользователе, о покупаемом товаре и о себе.

# **ПРОЕКТИРОВАНИЕ ЛОГИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ**

Логическая модель представляет собой модель базы данных (далее БД), которая не привязана к конкретной системе управления баз данных (далее СУБД). В ней выделяют основные объекты БД и определяют связи между этими объектами.

В ходе проектирования БД были выделены следующие сущности:

* Orders;
* Items;
* Catalogs;
* Users;
* Employees;
* ViewOrders.

Сущность «Orders» имеет следующие атрибуты:

* order\_id;
* user\_id;
* data;
* number;
* item\_id;
* employee\_id.

Сущность «Items» имеет следующие атрибуты:

* item\_id;
* name;
* price;
* quantity;
* score;
* description;
* catalog\_id.

Сущность «Catalogs» имеет следующие атрибуты:

* catalog\_id;
* name.

Сущность «Users» имеет следующие атрибуты:

* user\_id;
* first\_name;
* last\_name;
* patronymic;
* phone;
* email;
* url;
* status.
* Сущность «Employees» имеет следующие атрибуты:
* employee\_id;
* name;
* post.
* Сущность «ViewOrders» имеет следующие атрибуты:
* productname;
* description;
* price;
* namber;
* data;
* first\_name;
* last\_name;
* patronymic;
* phone;
* employees\_name;
* post;
* ordersum.

Связи между сущностями отображены на Рисунке 1.

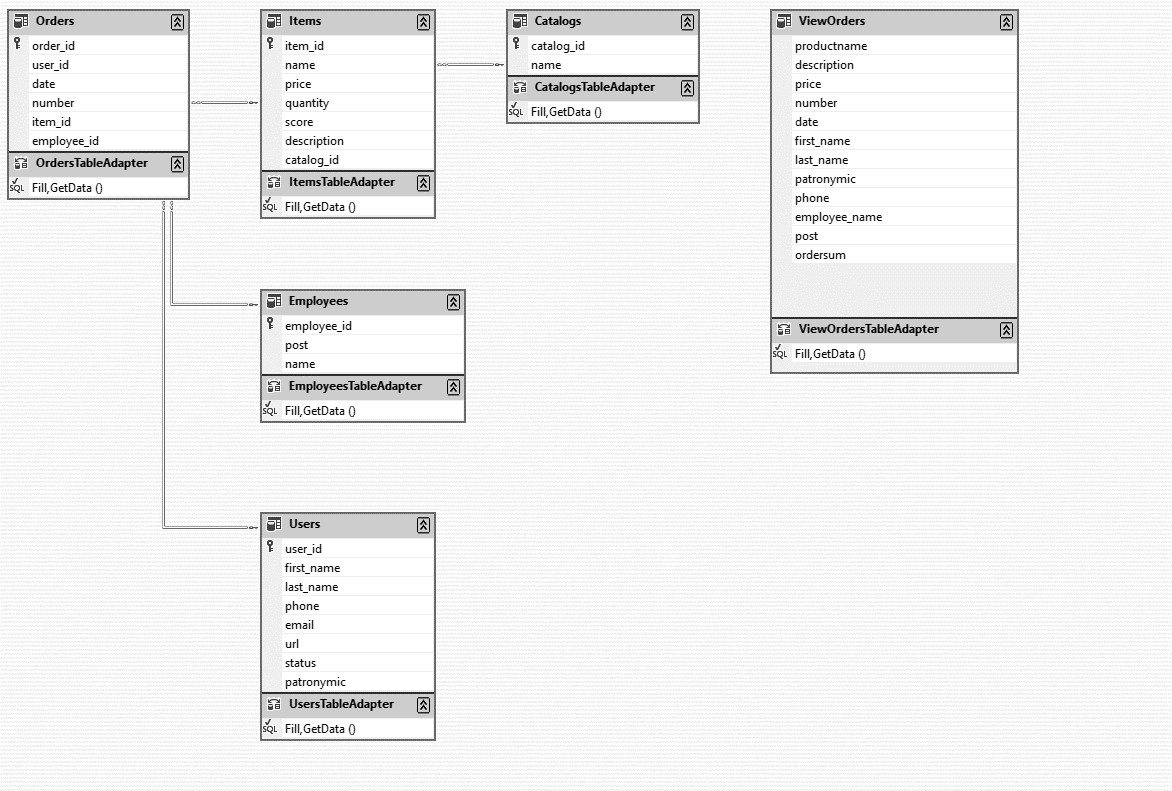


Рисунок 1 – Логическая модель БД

# **РАЗРАБОТКА ФИЗИЧЕСКОЙ МОДЕЛИ БАЗЫ ДАННЫХ**

Физическая модель – это модель данных, которая определяет, каким образом представляются данные, и содержит все детали, необходимые СУБД для создания базы данных.

Перед началом проектирования физической модели реляционная база данных должна быть нормализована. Процесс нормализации имеет цель устранения избыточности данных и заключается в приведении к третьей нормальной форме.

Первая нормальная форма (1НФ) требует, чтобы таблица была плоской и не содержала повторяющихся групп, иными словами, она не должна содержать ячеек, включающих несколько значений.

Вторая нормальная форма (2НФ) требует, чтобы все поля таблицы зависели от первичного ключа, то есть, чтобы первичный ключ однозначно

идентифицировал запись.

Для третьей нормальной формы (ЗНФ) требуется, чтобы все не ключевые столбцы таблицы зависели от первичного ключа таблицы, но были независимы друг от друга.

Для четвертой нормальной формы (4НФ) требуется, чтобы в одной таблице не содержались независимые элементы данных, если между ними существует отношение «многие-ко-многим».

Исходя из вышесказанного, база данных будет представлять собой пять связанных таблиц: Orders, item, catalog, employees, users, viewOrders.

Таблица 2 содержит информацию о таблицах, находящихся в составе БД

|  |  |  |  |
| --- | --- | --- | --- |
| Таблица | Атрибут, ограничение | Тип | Размер |
| Orders | order\_id (PK, FK) | Int | 4 байт |
| user\_id (NN) | Int | 4 байт |
| data (NN, FK) | date | 11 байт |
| number (NN, FK) | int | 11 байт |
| items\_id (NN) | Int | 4 байт |
| employee\_id (NN) | Int | 4 байт |
| Items | item\_id (PK, FK) | Int | 4 байт |
| name (NN) | VARCHAR | Max |
| price (NN) | float | 11 байт |
| quantyti (NN) | float | 11 байт |
| score (NN) | VARCHAR | 10 |
| description (NN) | VARCHAR | Max |
| catalog\_id (NN, FK) | Int | 4 байт |
| Catalogs | catalog\_id (PK, FK) | Int | 4 байт |
| name (NN) | VARCHAR | Max |
| Users | User\_id (PK, FK) | Int | 4 байт |
| first\_name (NN) | VARCHAR | 50 |
| last\_name(NN) | VARCHAR | 50 |
| patronymic (NN) | VARCHAR | 50 |
| phone (NN) | VARCHAR | 50 |
| email (NN) | VARCHAR | Max |
| url (NN) | VARCHAR | Max |
| status (NN) | VARCHAR | Max |
| Employees | employee\_id | Int |  |
| name | VARCHAR | 50 |
| post | VARCHAR | 50 |
| Viveworders | productname | VARCHAR | 50 |
| description | VARCHAR | Max |
| price | float | 11 байт |
| namber | Int | 11 байт |
| data | date | 11 байт |
| first\_name | VARCHAR | 50 |
| last\_name | VARCHAR | 50 |
| patronymic | VARCHAR | 50 |
| phone | VARCHAR | 50 |
| employee\_name | VARCHAR | 50 |
| post | VARCHAR | 50 |
| ordersum | VARCHAR | Max |

По данной таблице построим графически физическую модель.

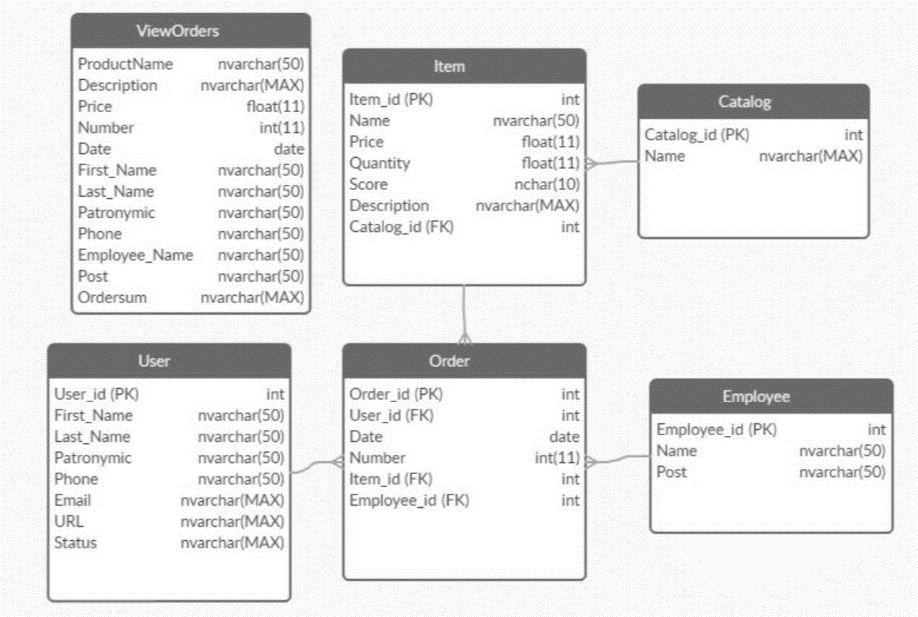


Рисунок 2 – Диаграмма физической модели

# **РАЗРАБОТКА ТЕХНИЧЕСКИХ ТРЕБОВАНИЙ К СЕРВЕРУ БАЗ ДАННЫХ. КОНФИГУРИРОВАНИЕ СЕТИ**

Выбор сервера пал на Microsoft SQL Server так как:

* СУБД масштабируется, поэтому работать с ней можно на портативных ПК или мощной мультипроцессорной технике. Процессор может одновременно обрабатывать большой объем запросов.
* Размер страниц – до 8 кб, поэтому данные извлекаются быстро, подробную и сложную информацию хранить удобнее. Система позволяет обрабатывать транзакции в интерактивном режиме, есть динамическая блокировка.
* Рутинные административные задачи автоматизированы: это управление блокировками, памятью, редактура размеров файлов. У системы продуманы настройки, можно создать профили пользователей.
* Реализован поиск по фразам, тексту, словам, можно создавать ключевые индексы.
* В SQL Server есть репликации через интернет, предусмотрена синхронизация. Есть полноценный веб-ассистент для форматирования страниц.
* В систему интегрирован сервер интерактивного анализа для принятия решений, создания корпоративных отчетов. Есть службы преобразования информации.
* Запросы можно формулировать на английском языке, без программирования.
* СУБД поддерживает работу с другими продуктами Microsoft: Access, Excel.

Системные требования к компьютеру:

* Жесткий диск

Для SQL Server требуется как минимум 6 ГБ свободного места на диске.

Требования к месту на диске определяются набором устанавливаемых компонентов SQL Server.

* Монитор

Для SQL Server требуется монитор Super VGA с разрешением 800x600 пикселей или более высоким.

* Интернет

Для поддержки функциональных средств Интернета требуется доступ к Интернету (могут применяться дополнительные тарифы).

* Память

Минимальные:

Экспресс-выпуски: 512 МБ

Все другие выпуски: 1 ГБ   
Рекомендуется:   
Экспресс-выпуски: 1 ГБ

Все другие выпуски: для обеспечения оптимальной производительности требуется не менее 4 ГБ с последующим увеличением по мере роста размера базы данных.

* Быстродействие процессора

Минимум: процессор x64 с тактовой частотой 1,4 ГГц  
Рекомендуется: 2,0 ГГц и выше

* Тип процессора     Процессор x64: AMD Opteron, AMD Athlon 64, Intel Xeon с поддержкой Intel EM64T, Intel Pentium IV с поддержкой EM64T.
* Требования к программному обеспечению   
  Операционная система

Windows 10 TH1 1507 или более поздней версии.  
Windows Server 2016 или более поздней версии.  
.NET Framework

Минимальная версия операционной системы подразумевает минимальную версию платформы .NET Framework.

* Сетевое программное обеспечение

Поддерживаемые операционные системы для SQL Server содержат встроенное сетевое программное обеспечение. Именованные экземпляры и экземпляры по умолчанию изолированной установки поддерживают следующие сетевые протоколы: Shared Memory, Named Pipes и TCP/IP.   
Программа установки SQL Server устанавливает следующие компоненты, необходимые для продукта:   
Собственный клиент SQL Server   
SQL Server Файлы поддержки программы установки

Поддержка операционных систем

В конфигурировании сети я выбрал топологию звезда.

Если вам нужно построить быструю и надёжную сеть, то отличный выбор — топология «звезда». «Кольцо» или «общая шина» также могут быть использованы на некоторых участках сети. Плюсы «звезды» — в её надёжности и простоте. К каждой рабочей станции идёт отдельный сетевой кабель, что весьма удобно и практично.

Благодаря этому в такой сети очень просто находить и исправлять неполадки, да и её обслуживание отнимает куда меньше времени и нервов. При подключении новых компьютеров к сети типа «звезда» она сохраняет свою работоспособность в отличие от других вариантов построения.

Например, топология «кольцо» не может похвастать подобной гибкостью.

Скорость в сети с топологией «звезда» ограничена лишь пропускной способностью кабеля и портов концентратора. Также в такой сети отсутствуют столкновения передаваемой информации.

# **РАЗРАБОТКА БАЗЫ ДАННЫХ И ТАБЛИЦ, ДОБАВЛЕНИЕ, УДАЛЕНИЕ ДАННЫХ**

При создании базы данных и таблицы, представленной в физической модели необходимо:

* Необходимо было перейти в SSMS (SQL Server Management Studio ) и подключился к серверу.
* Нажал ПКМ по папке база данных и выбрал создать базу данных.
* После открыл БД выбрал таблицы – создать - создать таблицу.
* При создании таблицы заносил имя, тип и ограничения столбцов таблицы.

Импорт данных:

1. Выбрали БД, в которую надо импортировать данные из таблицы и по ней нажали ПКМ – задачи – импорт данных…
2. Попали в мастера импорта и нажали далее.
3. Далее в источнике данных необходимо было выбрать excel и ниже надо было указать путь к файлу. После этого нажали далее.
4. После этого в назначении выбрали SQL Server Native Client 11.0 и сверили имя сервера и базу данных.
5. В следующем окне необходимо выбрать: скопировать данные из одной или нескольких таблиц, или представлений.
6. В этом окне необходимо было выбрать исходную таблицу.
7. В этом окне необходимо было убедиться в правильности выбора настроек и необходимо было нажать финиш.
8. В этом окне происходил процесс переноса данных в таблицу.
9. В итоге все данные из файла были перенесены в таблицу.

# **ОПИСАНИЕ МЕТОДОВ РЕВЬЮВИРОВАНИЯ**

Основой для ревьюирования кода служит техническая документация (ТД), основанная на ГОСТ 34 Разработка автоматизированной системы управления (АСУ).

Исходя из ГОСТ 34 Разработка автоматизированной системы управления, мы можем провести ревьюирование:

1. Проверка программы на соответствие ТЗ
   1. Название программы в ТЗ соответствует реальному названию программы
   2. Программа соответствует назначению и целям создания системы
   3. Оформление программы соответствует требованиям заказчика
   4. Код программы написан в соответствии с CamelCase, что отвечает требованиям заказчика
2. Проверка программы на работоспособность
   1. При нажатии кнопки Таблица Товарные группы, открывается соответствующая ей форма.
   2. При внесении данных в программе они вносятся в БД.
   3. При изменении данных в программе они изменяются в БД.
   4. При удалении данных они удаляются из БД.

После проведения ревьюирования программы нарушений не найдено.

# **ПРОВЕДЕНИЕ ТЕСТИРОВАНИЯ И ОТЛАДКИ РАЗРАБАТЫВАЕМЫХ ПРИЛОЖЕНИЙ**

При тестировании методом «черного ящика» тестировщик имеет доступ к ПО только через те же интерфейсы, что и заказчик или пользователь, либо через внешние интерфейсы, позволяющие другому компьютеру либо другому процессу подключиться к системе для тестирования. Группа тестирования изучает входные и выходные данные программного продукта. Ключом к успешному завершению системного тестирования является корректность данных. Поэтому на фазе создания данных для тестирования крайне важно, чтобы конечные пользователи предоставили как можно больше входных данных.

Черный ящик позволяет быстро выявить ошибки в функциональных спецификациях (в них описаны не только входные значения, но и то, что мы должны в итоге получить). Если полученный при тестировании результат отличается от заявленного в спецификации, то у нас появляется повод для общения с аналитиком для уточнения конечного результата.

Таблица 3 – Тестирование ИС методом «черного ящика»

|  |  |  |
| --- | --- | --- |
| Действие | Ожидаемый результат | Реальный результат |
| Ввести информацию о продавцах | Внесенная информация о продавцах | Внесенная информация о продавцах |
| Ввести информацию о товарных группах | Внесенная информация о товарных группах | Внесенная информация о товарных группах |
| Ввести информацию о новом поступлении товара | Внесенная информация о новом поступлении товара | Внесенная информация о новом поступлении товара |
| Сформировать заказ | Сформированный заказ | Сформированный заказ |
| Внесение изменений в заказ | Заказ с изменениями | Заказ с изменениями |
| Утвердить заказ и конечную стоимость | Утвержденный заказ с конечной стоимостью | Утвержденный заказ с конечной стоимостью |
| Ввести информацию о покупателе | Внесенная информация о покупателе | Внесенная информация о покупателе |

# **ЗАКЛЮЧЕНИЕ**

В ходе работы были получены навыки в ревьюирование программных продуктов. А также научился создавать триггеры, функции и представления. По мимо этого научился импортировать базу данных.

Таким образом, рабочая программа учебной практики УП.03 в рамках ПМ.03 Ревьюирование программных продуктов в период с 25.03.2022 по 29.03.2022 была выполнена мной в полном объеме.

При выполнении работ, предусмотренных заданием на практику, в организации ФГБОУ ВО «Вятский государственный университет» мной, Костровым Ильей Александровичем, соблюдались правила внутреннего трудового распорядка, охраны труда, техники безопасности, противопожарной безопасности, санитарно-эпидемиологические правила и гигиенические нормативы. Пройден вводный инструктаж и инструктаж на рабочем месте. Замечаний не имел.

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. ГОСТ Р 7.0.100-2018. Библиографическая запись. Библиографическое описание. Общие требования и правила составления: национальный стандарт Российской Федерации: дата введения 2019-07-01 / Федеральное агентство по техническому регулированию. – Изд. официальное. – Москва: Стандартинформ, 2018. – 124 с.
2. РАНХиГС: официальный сайт. – Москва. – Обновляется в течении суток. – URL: https://vlgr.ranepa.ru/about/struktura/podr/biblio/v-pomoshch-nauchnoy-rabote/obraztsy-po-gost-r-7-0-100-2018.php (дата обращения 27 марта 2022). – Текст: электронный.
3. Юридическая фирма Интернет и Право: официальный сайт. – Москва. – Обновляется в течении суток. – URL: https://internet-law.ru/gosts/gost/70535/ (дата обращения 27 марта 2022). – Текст: электронный.
4. MySlide.ru: официальный сайт. – Москва. – Обновляется в течении суток. – URL: https://myslide.ru/presentation/celi--zadachi--etapy-i-obekty-revyuirovaniya (дата обращения 24 марта 2022). – Текст: электронный.
5. КОД Журнал: официальный сайт. – Москва. – Обновляется в течении суток. – URL: https://thecode.media/code-review/ (дата обращения 25 марта 2022). – Текст: электронный.
6. Microsoft: официальный сайт. – Москва. – Обновляется в течении суток. – URL: https://docs.microsoft.com/ru-ru/visualstudio/ide/create-csharp-winform-visual-studio?view=vs-2019 (дата обращения 26 марта 2022). – Текст: электронный.
7. TECH: официальный сайт. – Санкт-Петербург. – Обновляется в течении суток. – URL: https://techrocks.ru/2019/02/14/code-review-ultimate-guide/ (дата обращения 25 марта 2022). – Текст: электронный.
8. iTechArt: официальный сайт. – Москва. – Обновляется в течении суток. – URL: https://www.itechart.by/blog-itechart/code-review-po-vsem-pravilam-chek-list-ot-razrabotchikov-itechart/ (дата обращения 25 марта 2022). – Текст: электронный.
9. vc.ru: официальный сайт. – Москва. – Обновляется в течении суток. – URL: https://vc.ru/dev/69792-organizovat-kod-revyu-sredi-tysyach-razrabotchikov-opyt-microsoft (дата обращения 26 марта 2022). – Текст: электронный.
10. Хабр: официальный сайт. – Санкт-Петербург. – Обновляется в течении суток. – URL: https://habr.com/ru/post/340550/ (дата обращения 25 марта 2022). – Текст: электронный.

# **ПРИЛОЖЕНИЕ А**

# **ЛИСТИНГ ПРОГРАММЫ**

**Users**

using System;

using System.Windows.Forms;

namespace Workings

{

public partial class Users : Form

{

public Users()

{

InitializeComponent();

}

private void usersBindingNavigatorSaveItem\_Click(object sender, EventArgs e)

{

this.Validate();

this.usersBindingSource.EndEdit();

this.tableAdapterManager.UpdateAll(this.computer\_shopDataSet);

}

private void Users\_Load(object sender, EventArgs e)

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "computer\_shopDataSet.Users". При необходимости она может быть перемещена или удалена.

this.usersTableAdapter.Fill(this.computer\_shopDataSet.Users);

}

}

}

**Orders**

using System;

using System.Windows.Forms;

namespace Workings

{

public partial class Orders : Form

{

public Orders()

{

InitializeComponent();

}

private void ordersBindingNavigatorSaveItem\_Click(object sender, EventArgs e)

{

this.Validate();

this.ordersBindingSource.EndEdit();

this.tableAdapterManager.UpdateAll(this.computer\_shopDataSet);

}

private void Orders\_Load(object sender, EventArgs e)

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "computer\_shopDataSet.ViewOrders". При необходимости она может быть перемещена или удалена.

this.viewOrdersTableAdapter.Fill(this.computer\_shopDataSet.ViewOrders);

// TODO: данная строка кода позволяет загрузить данные в таблицу "computer\_shopDataSet.Orders". При необходимости она может быть перемещена или удалена.

this.ordersTableAdapter.Fill(this.computer\_shopDataSet.Orders);

}

}

}

**ItemsTableViewForm**

using System;

using System.Drawing;

using System.Windows.Forms;

namespace Workings

{

public partial class ItemsTableViewForm : Form

{

private System.Windows.Forms.DataGridViewColumn collumnName;

public ItemsTableViewForm()

{

InitializeComponent();

}

private void itemsBindingNavigatorSaveItem\_Click(object sender, EventArgs e)

{

this.Validate();

this.itemsBindingSource.EndEdit();

this.tableAdapterManager.UpdateAll(this.computer\_shopDataSet);

}

private void ItemsTableViewForm\_Load(object sender, EventArgs e)

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "computer\_shopDataSet.Items". При необходимости она может быть перемещена или удалена.

this.itemsTableAdapter.Fill(this.computer\_shopDataSet.Items);

}

private void sortlistBox\_SelectedIndexChanged(object sender, EventArgs e)

{

sortBtn.Enabled = true;

}

private void sortBtn\_Click(object sender, EventArgs e)

{

//создание переменной для хранения имени выбранного столбца

this.collumnName = new System.Windows.Forms.DataGridViewColumn();

switch (sortlistBox.SelectedIndex)

{

case 0:

collumnName = nameDataGridViewTextBoxColumn;

break;

case 1:

collumnName = priceDataGridViewTextBoxColumn;

break;

case 2:

collumnName = countDataGridViewTextBoxColumn;

break;

case 3:

collumnName = markDataGridViewTextBoxColumn;

break;

case 4:

collumnName = descDataGridViewTextBoxColumn;

break;

case 5:

collumnName = productGroupDataGridViewTextBoxColumn;

break;

}

if (sortUpRadioBtn.Checked)

{

itemsDataGridView.Sort(collumnName, System.ComponentModel.ListSortDirection.Ascending);

}

else

{

itemsDataGridView.Sort(collumnName, System.ComponentModel.ListSortDirection.Descending);

}

}

private void filterBtn\_Click(object sender, EventArgs e)

{

itemsBindingSource.Filter = $"NAME='{nameComboBox.Text}'";

}

private void showAllBtn\_Click(object sender, EventArgs e)

{

itemsBindingSource.Filter = "";

}

private void findBtn\_Click(object sender, EventArgs e)

{

for (int i = 0; i < itemsBindingSource.Count - 1; i++)

{

for (int j = 0; j < itemsBindingSource.Count - 1; j++)

{

itemsDataGridView[i, j].Style.BackColor = Color.White;

itemsDataGridView[i, j].Style.ForeColor = Color.Black;

}

}

for (int i = 0; i < itemsDataGridView.ColumnCount - 1; i++)

{

for (int j = 0; j < itemsDataGridView.RowCount - 1; j++)

{

if (itemsDataGridView[i, j].Value.ToString().IndexOf(criteriaTextBox.Text) != -1)

{

itemsDataGridView[i, j].Style.BackColor = Color.AliceBlue;

itemsDataGridView[i, j].Style.ForeColor = Color.Blue;

}

}

}

}

private void closeBtn\_Click(object sender, EventArgs e)

{

this.Close();

}

}

}

**Items**

using System;

using System.Windows.Forms;

namespace Workings

{

public partial class Items : Form

{

private ItemsTableViewForm itemsTableViewForm;

public Items()

{

InitializeComponent();

}

private void itemsBindingNavigatorSaveItem\_Click(object sender, EventArgs e)

{

this.Validate();

this.itemsBindingSource.EndEdit();

this.tableAdapterManager.UpdateAll(this.computer\_shopDataSet);

}

private void Items\_Load(object sender, EventArgs e)

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "computer\_shopDataSet.Orders". При необходимости она может быть перемещена или удалена.

this.ordersTableAdapter.Fill(this.computer\_shopDataSet.Orders);

// TODO: данная строка кода позволяет загрузить данные в таблицу "computer\_shopDataSet.Catalogs". При необходимости она может быть перемещена или удалена.

this.catalogsTableAdapter.Fill(this.computer\_shopDataSet.Catalogs);

// TODO: данная строка кода позволяет загрузить данные в таблицу "computer\_shopDataSet.Items". При необходимости она может быть перемещена или удалена.

this.itemsTableAdapter.Fill(this.computer\_shopDataSet.Items);

}

private void buttonNext\_Click(object sender, EventArgs e)

{

itemsBindingSource.MoveFirst();

}

private void buttonPrevious\_Click(object sender, EventArgs e)

{

itemsBindingSource.MovePrevious();

}

private void buttonNext\_Click\_1(object sender, EventArgs e)

{

itemsBindingSource.MoveNext();

}

private void buttonLast\_Click(object sender, EventArgs e)

{

itemsBindingSource.MoveLast();

}

private void buttonAdd\_Click(object sender, EventArgs e)

{

itemsBindingSource.AddNew();

}

private void buttonDelete\_Click(object sender, EventArgs e)

{

itemsBindingSource.RemoveCurrent();

}

private void buttonSave\_Click(object sender, EventArgs e)

{

this.Validate();

this.itemsBindingSource.EndEdit();

this.tableAdapterManager.UpdateAll(this.computer\_shopDataSet);

}

private void openTableViewBtn\_Click(object sender, EventArgs e)

{

itemsTableViewForm = new ItemsTableViewForm();

itemsTableViewForm.Visible = true;

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

}

}

}

**FormMain**

using System;

using System.Windows.Forms;

namespace Workings

{

public partial class FormMain : Form

{

private Users users;

private Catalogs catalogs;

private Orders orders;

private Items items;

public FormMain()

{

InitializeComponent();

}

private void buttonUsers\_Click(object sender, EventArgs e)

{

users = new Users();

users.Visible = true;

}

private void buttonCategory\_Click(object sender, EventArgs e)

{

catalogs = new Catalogs();

catalogs.Visible = true;

}

private void buttonOrders\_Click(object sender, EventArgs e)

{

orders = new Orders();

orders.Visible = true;

}

private void buttonItems\_Click(object sender, EventArgs e)

{

items = new Items();

items.Visible = true;

}

private void FormMain\_Load(object sender, EventArgs e)

{

}

}

}

**Catalogs**

using System;

using System.Windows.Forms;

namespace Workings

{

public partial class Catalogs : Form

{

public Catalogs()

{

InitializeComponent();

}

private void catalogsBindingNavigatorSaveItem\_Click(object sender, EventArgs e)

{

this.Validate();

this.catalogsBindingSource.EndEdit();

this.tableAdapterManager.UpdateAll(this.computer\_shopDataSet);

}

private void Catalogs\_Load(object sender, EventArgs e)

{

// TODO: данная строка кода позволяет загрузить данные в таблицу "computer\_shopDataSet.Catalogs". При необходимости она может быть перемещена или удалена.

this.catalogsTableAdapter.Fill(this.computer\_shopDataSet.Catalogs);

}

}

}

# **ПРИЛОЖЕНИЕ Б**

# **ФОРМЫ ПРОЕКТА И IDF ДИАГРАММА**

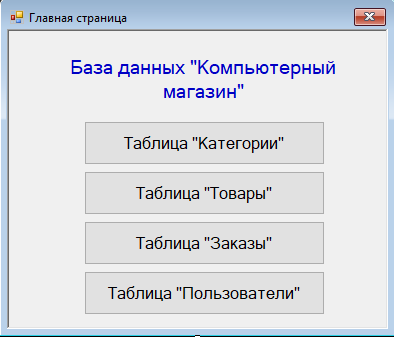


Рисунок 3 – меню

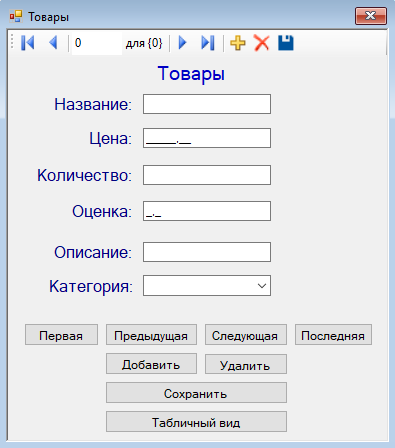


Рисунок 4 - товары

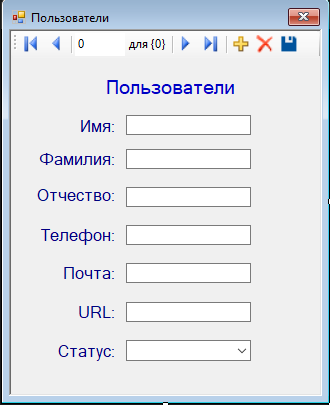


Рисунок 5 – пользователи

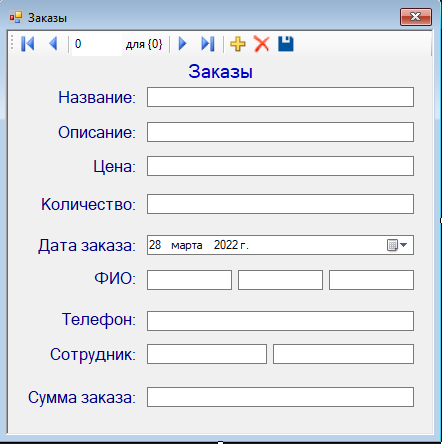


Рисунок 6 – заказы

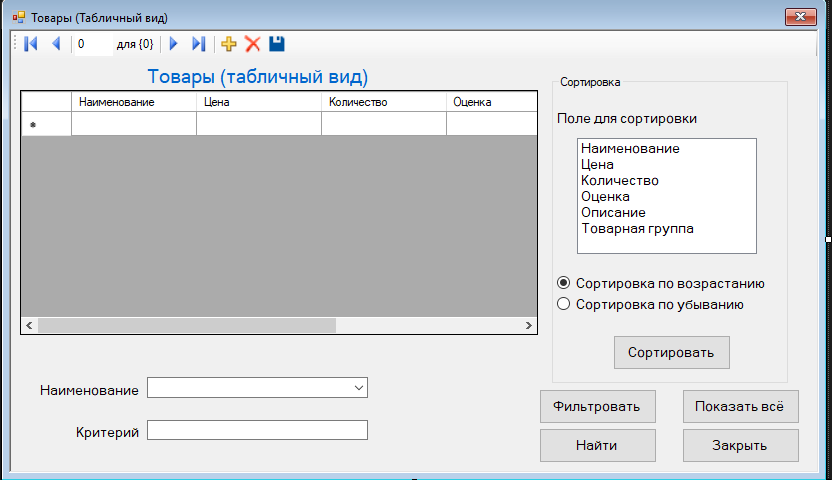


Рисунок 7 – табличный вид

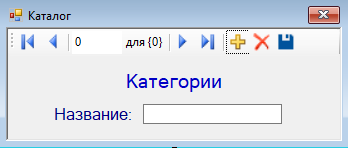


Рисунок 8 – каталог

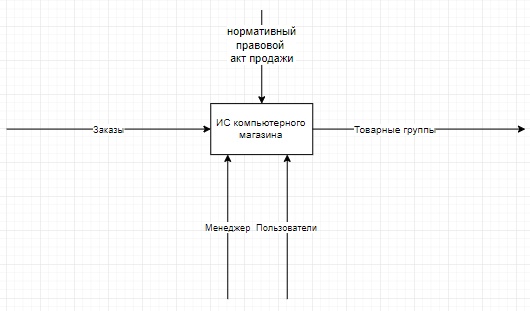


Рисунок 9 – Idf0

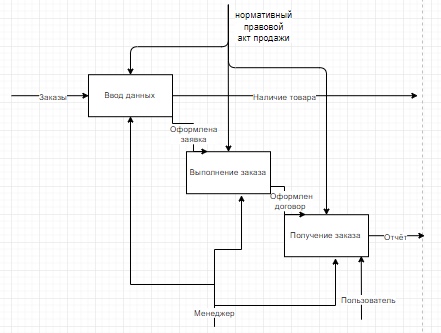


Рисунок 10 – idf1

# **ПРИЛОЖЕНИЕ В**

# **ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

1. Общие сведения
   1. Наименование системы
      1. Полное наименование: Computer Shop
      2. Краткое наименование: CSH
   2. Основания для проведения работ

Работа выполняется на основании договора №1/12-11-11-001 от 11.11.2020

* 1. Наименование организации – Заказчика и Разработчика
     1. Заказчик

Заказчик: ООО Магазин компьютерной техники и периферии

Адрес фактический: г. Киров, ул. Карла Маркса, д.76

Телефон: +7(953) 22222269

* + 1. Разработчик

Разработчик: ООО компания КБ

Адрес фактический: г. Кирово, ул Спасская, д 29

* 1. Плановые сроки начала и окончания работ

Плановый срок начала работ по созданию Системы автоматизации для компьютерного магазина «Computer Group» — 23 марта 2022.

Плановый срок окончания работ по созданию Системы автоматизации для компьютерного магазина «Computer Group» — 29 марта 2022.

* 1. Порядок оформления и предъявления заказчику результатов работ

Работы по созданию CSH Компьютерный магазин сдаются Разработчиком поэтапно в соответствии с календарным планом Проекта. По окончании каждого из этапов работ Разработчик сдает Заказчику соответствующие отчетные документы этапа, состав которых определены Договором.

1. Назначения и цели создания системы
   1. Назначение системы

Компьютерный магазин предназначен для повышения оперативности и качества принимаемых управленческих решений сотрудниками Заказчика.

* 1. Цели создания системы

Главной целью CSH «Компьютерный магазин» является:

* предоставления людям недорогой качественной компьютерной периферии
* повышения качества (полноты, точности, достоверности, своевременности, согласованности) информации.