МИНИСТЕРСТВО НАУКИ и высшего образования РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования «МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ»

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

филиал «РКТ» МАИ в г. Химки Московской области

**Специальность 09.02.07 «Информационные системы и программирование»**

**ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

**ПМ.04 «Разработка, администрирование и защита баз данных»**

**Студент**

**группы ИСП-31-19 /Бойцова В.И.**

**Руководитель**

**практики от филиала /**

**2021г.**

**ПРОГРАММА УЧЕБНОЙ ПРАКТИКИ**

По специальности 09.02.07 «Информационные системы и программирование»

|  |  |  |
| --- | --- | --- |
| Дата | Наименование выполняемых работ | Количество дней практики |
|  | Ознакомление с программой практики. Прохождение инструктажа по технике безопасности и охране труда, изучение внутреннего распорядка организации и правил работы. Разработка задания на производственную практику. |  |
|  | Описание структуры сети, в которой может функционировать разработанная база данных. |  |
|  | Описание сетевого оборудования необходимого для работы с базой данных. |  |
|  | Описание программного обеспечения необходимого для работы с базой данных по сети. |  |
|  | Разработка клиентских интерфейсов для клиент-серверных приложений. |  |
|  | Создание инфокоммуникационной системы.  Выполнение сетевых настроек для взаимодействия с СУБД. |  |
|  | Создание концептуальной модели данных.  Создание логической модели данных.  Создание физической модели данных.  Описание механизмов обеспечения целостности базы данных. |  |
|  | Описание СУБД представленной базы данных и ее возможностей. Внесение различных данных в базу данных.  Структурирование запросов базы данных.  Определение методов создания хранимых процедур и триггеров. Создание хранимых процедур и триггеров базы данных. |  |
|  | Определение способов управления правами пользователей.  Описание распределения прав пользователей и управления ими в базе данных. |  |
|  | Определение методов создания и синхронизации реплик базы данных. Описание существующих механизмов репликации в базе данных. Определение методов создания резервных копий базы данных. Описание существующих механизмов резервного копирования в базе данных. |  |
|  | Составление отчёта по практике |  |
|  | Подготовка к защите отчёта по практике. |  |
|  | Итоговая аттестация по учебной практике - зачёт |  |

Руководитель практики от филиала «РКТ» МАИ в г. Химки Московской области

(должность, фамилия, имя, отчество)

|  |  |  |
| --- | --- | --- |
| Дата | *(подпись)* | *Расшифровка подписи* |

**АТТЕСТАЦИОННЫЙ ЛИСТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

Студент Бойцова Вероника Игоревна , обучающийся на 3 курсе по специальности СПО 09.02.07 Информационные системы и программирование успешно прошел(ла) учебную практику по профессиональному модулю ПМ.04 «Разработка, администрирование и защита баз данных» в объеме часов часов с « » 2022г. по « » 2022г.

В филиале «РКТ» МАИ в г. Химки Московской области

*Наименование предприятия/организации, юридический адрес*

**Виды и качество выполнения работ**

|  |  |  |
| --- | --- | --- |
| Виды работ, выполненных обучающимся во время практики | Объём работ  (час.) | Качество выполнения работ в соответствии с технологией и (или) требованиями организации, в которой походила практика |
| Ознакомление с программой практики. Прохождение инструктажа по технике безопасности и охране труда, изучение внутреннего распорядка организации и правил работы. Разработка задания на производственную практику. |  |  |
| Описание структуры сети, в которой может функционировать разработанная база данных. |  |  |
| Описание сетевого оборудования необходимого для работы с базой данных. |  |  |
| Описание программного обеспечения необходимого для работы с базой данных по сети. |  |  |
| Разработка клиентских интерфейсов для клиент-серверных приложений. |  |  |
| Создание инфокоммуникационной системы.  Выполнение сетевых настроек для взаимодействия с СУБД. |  |  |
| Создание концептуальной модели данных.  Создание логической модели данных.  Создание физической модели данных.  Описание механизмов обеспечения целостности базы данных. |  |  |
| Описание СУБД представленной базы данных и ее возможностей. Внесение различных данных в базу данных.  Структурирование запросов базы данных.  Определение методов создания хранимых процедур и триггеров. Создание хранимых процедур и триггеров базы данных. |  |  |
| Определение способов управления правами пользователей.  Описание распределения прав пользователей и управления ими в базе данных |  |  |
| Определение методов создания и синхронизации реплик базы данных. Описание существующих механизмов репликации в базе данных. Определение методов создания резервных копий базы данных. Описание существующих механизмов резервного копирования в базе данных. |  |  |
| Составление отчёта по практике |  |  |
| Подготовка к защите отчёта по практике. |  |  |
| Итоговая аттестация по учебной практике - зачёт |  |  |

Руководитель практики от филиала «РКТ» МАИ в г. Химки Московской области

(должность, фамилия, имя, отчество)

|  |  |  |
| --- | --- | --- |
| Дата | *(подпись)* | *Расшифровка подписи* |

**ХАРАКТЕРИСТИКА**

Обучающаяся \_\_ Бойцова В.И.\_\_\_\_\_\_\_\_\_\_ группы ИСП-31-19 по специальности СПО 09.02.07 Информационные системы и программирование успешно прошла производственную практику по профессиональному модулю ПМ.04 «Разработка, администрирование и защита баз данных» в объеме часов с « »   2022г. по « » 2022г.

в организации филиал «РКТ» МАИ в г. Химки Московской области

*Наименование предприятия /организации, юридический адрес*

За время работы проявила себя как ответственный/безответственный, коммуникабельный/замкнутый, исполнительный/неисполнительный, дисциплинированный/имеет дисциплинарные замечания, доброжелательный/наглый сотрудник.

Обучающаяся обладает общими компетенциями, включающими в себя способность:

ПК 11.1. Осуществлять сбор, обработку и анализ информации для проектирования баз данных.

ПК 11.2. Проектировать базу данных на основе анализа предметной области.

ПК 11.3. Разрабатывать объекты базы данных в соответствии с результатами анализа предметной

области.

ПК 11.4. Реализовывать базу данных в конкретной системе управления базами данных.

ПК 11.5. Администрировать базы данных.

ПК 11.6. Защищать информацию в базе данных с использованием технологии защиты информации.

К работе относился \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Цели и задачи практики достигнуты/ достигнуты не в полном объеме.

Оценка за практику\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

Руководитель практики от филиала «РКТ» МАИ в г. Химки Московской области

(должность, фамилия, имя, отчество)

|  |  |  |
| --- | --- | --- |
| Дата | *(подпись)* | *Расшифровка подписи* |

**ДНЕВНИК ПРОХОЖДЕНИЯ ПРАКТИКИ**

|  |  |  |
| --- | --- | --- |
| Дата | Наименование выполняемых работ | Подпись руководителя предприятия/ОУ |
|  | Ознакомление с программой практики. Прохождение инструктажа по технике безопасности и охране труда, изучение внутреннего распорядка организации и правил работы. Разработка задания на производственную практику. |  |
|  | Описание структуры сети, в которой может функционировать разработанная база данных. |  |
|  | Описание сетевого оборудования необходимого для работы с базой данных. |  |
|  | Описание программного обеспечения необходимого для работы с базой данных по сети. |  |
|  | Разработка клиентских интерфейсов для клиент-серверных приложений. |  |
|  | Создание инфокоммуникационной системы.  Выполнение сетевых настроек для взаимодействия с СУБД. |  |
|  | Создание концептуальной модели данных.  Создание логической модели данных.  Создание физической модели данных.  Описание механизмов обеспечения целостности базы данных. |  |
|  | Описание СУБД представленной базы данных и ее возможностей. Внесение различных данных в базу данных.  Структурирование запросов базы данных.  Определение методов создания хранимых процедур и триггеров. Создание хранимых процедур и триггеров базы данных. |  |
|  | Определение способов управления правами пользователей.  Описание распределения прав пользователей и управления ими в базе данных. |  |
|  | Определение методов создания и синхронизации реплик базы данных. Описание существующих механизмов репликации в базе данных. Определение методов создания резервных копий базы данных. Описание существующих механизмов резервного копирования в базе данных. |  |
|  | Составление отчёта по практике |  |
|  | Подготовка к защите отчёта по практике. |  |
|  | Итоговая аттестация по учебной практике - зачёт |  |

**Параметры учета выполнения работ по заданию**

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Наименование вида работ | Количество дней практики | Прилагаемый материал в отчете |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 7](#_Toc107304531)

[1 Подготовка к разработке базы данных 8](#_Toc107304532)

[1.1 Анализ актуальности 8](#_Toc107304533)

[1.2 Постановка задачи 8](#_Toc107304534)

[1.3 Анализ инструментальных средств 8](#_Toc107304535)

[2 Проектирование БД 12](#_Toc107304536)

[2.1 Концептуальная модель 12](#_Toc107304537)

[2.2 Логическая модель 15](#_Toc107304538)

[2.3 Физическая модель 17](#_Toc107304539)

[3 Выбор СУБД. Разработка и настройка Базы Данных 18](#_Toc107304540)

[3.1 Сравнение СУБД 18](#_Toc107304541)

[3.2 Разработка базы данных 18](#_Toc107304542)

[4. Создание приложения 18](#_Toc107304543)

[4.1 Разработка шаблонов и макетов 18](#_Toc107304544)

[4.2 Разработка интерфейсов 18](#_Toc107304545)

[4.3 организация связей с БД 18](#_Toc107304546)

[ЗАКЛЮЧЕНИЕ 19](#_Toc107304547)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 20](#_Toc107304548)

[ПРИЛОЖЕНИЕ А. SQL запросы 21](#_Toc107304549)

[1. Запросы на создание таблиц 21](#_Toc107304550)

[2. Запросы на добавление данных 23](#_Toc107304551)

ВВЕДЕНИЕ

Производственная практика является важной составной частью учебного процесса подготовки высококвалифицированных специалистов. В период прохождения практики студенты конкретизируют и закрепляют теоретические знания, приобретенные в процессе обучения, овладевают навыками практической работы, стараются применить полученные знания на практике. Практика является завершающим этапом в процессе подготовки специалиста к самостоятельной производственной деятельности.

Целью практики является разработка базы данных для интернет-магазина продуктов. Так же необходимо разработать приложение для работы с базой данных.

Для достижения поставленной цели необходимо выполнить следующие задачи:

1. Разработка базы данных интернет-магазина продуктов, создание его концептуальной, логической и физической моделей.
2. Разработка приложения для работы с базой данных и организация взаимодействия.
3. Создание исполняемого файла для работы приложения, создание инсталлятора для его установки на других устройствах.
4. Создание отчетной документации.

1 Подготовка к разработке базы данных

## **1.1 Анализ актуальности**

Создание базы данных для интернет-магазина продуктов является актуальным, поскольку периодически открываются новые интернет-магазины, предоставляющие возможность покупки различных товаров, в том числе и конкретно продуктов. Наличие уже готовых примеров приложений, позволяющих создать необходимую базу данных и взаимодействовать с ней, значительно упростит процесс создания и работы интернет-магазина.

## **1.2 Постановка задачи**

Постановка задачи является важным этапом, позволяющим конкретизировать и четко поставить как финальную задачу, так и этапы её достижения.

Главной задачей является разработка базы данных для интернет-магазина продуктов, приложения для удобного взаимодействия с базой данных. Создать исполнительный файл программы и установщик для возможности установки программы на других компьютерах.

Основными задачами являются:

1. Исследование предметной области. Исследование и обоснование выбора инструментальных средств. Оценка актуальности поставленной задачи.
2. Разработка базы данных интернет-магазина продуктов, его концептуальной, логической и физической моделей.
3. Разработка приложения-интерфейса для работы с базой данных с возможностью использования персоналом с различными должностями.
4. Создание исполняемого файла для работы приложения, создание инсталлятора для его установки на других устройствах.

## **1.3 Анализ инструментальных средств**

Для разработки базы данных необходимо выбрать СУБД. При выборе СУБД были рассмотрены следующие: MySQL и Microsoft SQL сервер.

|  |  |  |
| --- | --- | --- |
| Название СУБД | Положительные стороны | Отрицательные стороны |
| MySQL | Распространяется бесплатно;  Большое количество документации;  Предлагает крупный функционал даже в бесплатной версии;  Поддерживает набор пользовательских интерфейсов;  Может работать с другими СУБД;  Взаимодействует с любым языком программирования. | Затратен по времени и усилиям даже для простых задач;  Отсутствует встроенная поддержка XML или OLAP;  Бесплатная версия имеет только платную поддержку. |
| Microsoft SQL сервер | Прост в использовании;  Работает быстро и стабильно;  Возможность регулировать потребляемые ресурсы;  Доступ к визуализации на мобильных устройствах;  Хорошо взаимодействует с другими продуктами Microsoft. | Высокая цена для юридических лиц;  Может использовать все ресурсы;  Есть проблемы с интеграцией для импорта файлов;  Смысл для покупки лицензии только при наличии всей экосистемы Microsoft. |

Поскольку MySQL взаимодействует с многими языками программирования, была выбрана она.

Для разработки приложения взаимодействия необходимо выбрать подходящий язык программирования. Из существующих языков на рассмотрение были выбраны два распространенных языка: C++ и Python.

|  |  |  |
| --- | --- | --- |
| Название языка | Положительные стороны | Отрицательные стороны |
| C++ | Высокая скорость работы итоговых программ;  Прямое управление динамической памятью;  Востребованность;  Универсальность использования. | Сложность освоения в связи с трудным для понимания синтаксисом;  Необходимость внимательно работать с указателями, ссылками, памятью. |
| Python | Простота освоения; Лаконичность;  Легкий и понятный синтаксис;  Интерпретируемость;  Востребованность;  Динамическая типизация;  Большое количество библиотек;  Большое количество документации и учебных материалов;  Кроссплатформенность. | Медленная скорость выполнения программ;  Программы, разработанные на Python требуют большое количество памяти;  При компиляции не выполняется проверка, что ведет к появлению ошибок и требует очень качественное тестирование. |

Из представленных языков для работы был выбран Python в связи с сравнительной простотой освоения и разработки программы.

Для работы так же необходимо выбрать среду разработки. Так как используется язык Python, то и среду стоит выбрать для него подходящую.

|  |  |  |
| --- | --- | --- |
| Название среды | Положительные стороны | Отрицательные стороны |
| Atom | Кроссплатформенность;  Легкость скачивания и установки. | Поддержка сборки и отладки добавляется с помощью расширений;  Работает как JavaScript-процесс, а не как нативное приложение. |
| Visual Studio | Наличие полноценной бесплатной версии программы;  Много качественных плагинов, расширяющих возможности приложения;  Поддержка платформы .NET;  Облачные хранилища;  Возможность работы в команде. | Сложность самостоятельного освоения;  Требовательность к ресурсам,  Проблемы при переходе на платную версию. |
| PyСharm | Разработан специально для Python;  Поддержка виртуальной среды Anaconda; Кроссплатформенность;  Наличие полноценной бесплатной версии программы. | Тормозит на устройствах, имеющих до 8Гб оперативной памяти. |

Для разработки выбрана среда PyСharm, так как является разработанной специально для работы с Python.

2 Проектирование БД

Для создания концептуальной и логической моделей базы в качестве вспомогательного средства была использована программа ERDPlus.

В моделях требовалось отразить следующие сущности:

1. Клиент;
2. Сотрудник;
3. Компания;
4. Продукт;
5. Заказ;
6. Список покупок.

## **2.1 Концептуальная модель**

Концептуальная модель представлена на рисунке 1.

Модель описывает то, как должны будут взаимодействовать между собой сущности, а так же описывает их атрибуты.

Каждой из сущностей присвоен ряд атрибутов, отображающих свойства относящихся к ней объектов:

Для сущности Клиент описаны следующие атрибуты:

1. ID клиента — первичный ключ;
2. Имя;
3. Фамилия;
4. E-mail;
5. Телефон;
6. Страна;
7. Город;
8. Улица;
9. Почтовый индекс;
10. Логин;
11. Пароль.

Для сущности Сотрудник описаны следующие атрибуты:

1. ID сотрудника — первичный ключ;
2. Имя;
3. Фамилия;
4. Должность;
5. Телефон;
6. E-mail;
7. Логин;
8. Пароль.

Для сущности Компания описаны следующие атрибуты:

1. ID компании — первичный ключ;
2. Название;
3. Телефон;
4. E-mail.

Для сущности Продукт описаны следующие атрибуты:

1. ID продукта — первичный ключ;
2. Название;
3. Цена;
4. Количество;
5. Код компании — внешний ключ, связывающий с сущностью Компания.

Для сущности Заказ описаны следующие атрибуты:

1. ID заказа — первичный ключ;
2. Почтовый индекс;
3. Страна;
4. Город;
5. Улица;
6. Имя клиента;
7. Код клиента — внешний ключ, связывающий с сущностью Клиент;
8. Код сотрудника — внешний ключ, связывающий с сущностью Сотрудник.

Для сущности Список покупок описаны следующие атрибуты:

1. ID элемента списка — первичный ключ;
2. Цена;
3. Количество;
4. Код заказа — внешний ключ, связывающий с сущностью Заказ;
5. Код продукта — внешний ключ, связывающий с сущностью Продукт.

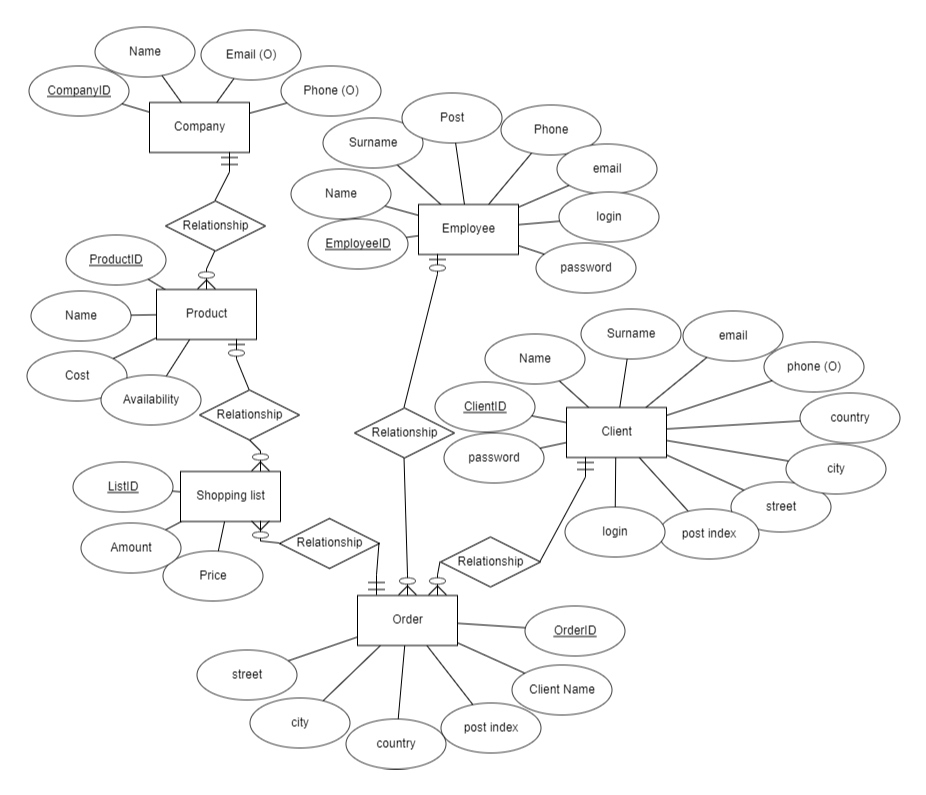


Рисунок 1 — Концептуальная модель.

Сущности имеют следующие связи:

Связь один ко многим между сущностью клиента и сущностью заказа, так как один клиент может иметь более одного заказа. При этом у заказа обязательно должен быть заказчик (клиент).

Связь один ко многим между сущностью сотрудника и сущностью заказа, так как один сотрудника может управлять более чем одним заказом. При этом у заказа обязательно должен быть тот, кто его обслужит (сотрудник).

Связь один ко многим между сущностью компании и сущностью продукта, так как одна компания может иметь в списке производимых продуктов более одного продукта. При этом у товара обязательно должен быть тот, кто его поставляет (компания).

Связь один ко многим между сущностью заказа и сущностью списка покупок, так как в одном заказе может быть более одного продукта. При этом в заказе должен быть хотя бы один продукт.

Связь один ко многим между сущностью продукт и сущностью списка продуктов, так как в списке продуктов один продукт может встречаться несколько раз.

## **2.2 Логическая модель**

Логическая модель представлена на рисунке 2.

Логическая модель так же описывает атрибуты сущностей и их связи. При этом логическая модель так же описывает типы данных, присущие каждому атрибуту.

Для сущности Клиент атрибутам присвоены следующие типы данных:

1. ID клиента — первичный ключ — INT auto\_increment (автоматичски присваиваемый номер);
2. Имя — VARCHAR(100) NOT NULL;
3. Фамилия — VARCHAR(100) NOT NULL;
4. E-mail — VARCHAR(100) NOT NULL;
5. Телефон — VARCHAR(11) NOT NULL;
6. Страна — VARCHAR(100) NOT NULL;
7. Город — VARCHAR(100) NOT NULL;
8. Улица — VARCHAR(100) NOT NULL;
9. Почтовый индекс — VARCHAR(10) NOT NULL;
10. Логин — VARCHAR(100) NOT NULL;
11. Пароль — VARCHAR(50) NOT NULL.

Для сущности Сотрудник атрибутам присвоены следующие типы данных:

1. ID сотрудника — первичный ключ — INT auto\_increment (автоматичски присваиваемый номер);
2. Имя — VARCHAR(100) NOT NULL;
3. Фамилия — VARCHAR(100) NOT NULL;
4. Должность — VARCHAR(100) NOT NULL;
5. Телефон — VARCHAR(11) NOT NULL;
6. E-mail — VARCHAR(100) NOT NULL;
7. Логин — VARCHAR(100) NOT NULL;
8. Пароль — VARCHAR(50) NOT NULL.

Для сущности Компания атрибутам присвоены следующие типы данных:

1. ID компании — первичный ключ — INT auto\_increment (автоматичски присваиваемый номер);
2. Название — VARCHAR(100) NOT NULL;
3. Телефон — VARCHAR(11);
4. E-mail — VARCHAR(100).

Для сущности Продукт атрибутам присвоены следующие типы данных:

1. ID продукта — первичный ключ — INT auto\_increment (автоматичски присваиваемый номер);
2. Название — VARCHAR(100) NOT NULL;
3. Цена — INT NOT NULL;
4. Количество — INT NOT NULL;
5. Код компании — INT NOT NULL — внешний ключ, связывающий с сущностью Компания.

Для сущности Заказ атрибутам присвоены следующие типы данных:

1. ID заказа — первичный ключ — INT auto\_increment (автоматичски присваиваемый номер);
2. Почтовый индекс — VARCHAR(10) NOT NULL;
3. Страна — VARCHAR(100) NOT NULL;
4. Город — VARCHAR(100) NOT NULL;
5. Улица — VARCHAR(100) NOT NULL;
6. Имя клиента — VARCHAR(100) NOT NULL;
7. Код клиента — INT NOT NULL — внешний ключ, связывающий с сущностью Клиент;
8. Код сотрудника — INT NOT NULL — внешний ключ, связывающий с сущностью Сотрудник.

Для сущности Список покупок атрибутам присвоены следующие типы данных:

1. ID элемента списка — первичный ключ — INT auto\_increment (автоматичски присваиваемый номер);
2. Цена — INT NOT NULL;
3. Количество — INT NOT NULL;
4. Код заказа — INT NOT NULL — внешний ключ, связывающий с сущностью Заказ;
5. Код продукта — INT NOT NULL — внешний ключ, связывающий с сущностью Продукт.

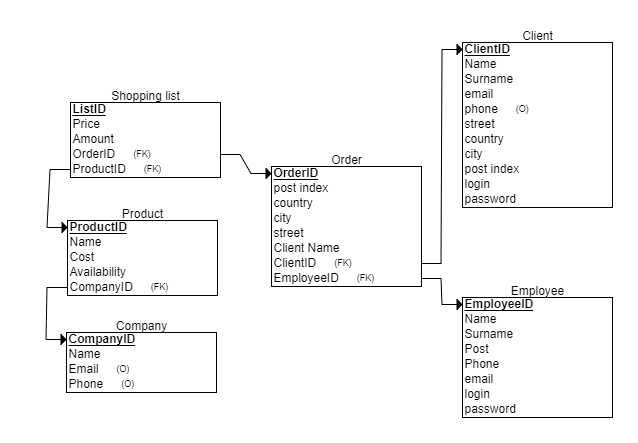


Рисунок 2 — Логическая модель.

## **2.3 Физическая модель**

Физической моделью является непосредственная реализация с помощью какой-либо СУБД. В данной работе реализация базы данных сделана с помощью СУБД MySQL.

3 Выбор СУБД. Разработка и настройка Базы Данных

## **3.1 Сравнение СУБД**

Сравнение СУБД описано в разделе 1.3 Анализ инструментальных средств.

## **3.2 Разработка базы данных**

Создание SQL запросов на основе логической модели, описанной в разделе 2.2 логическая модель. SQL запросы на создание таблиц и добавление некоторых данных представлены в приложении А.

Создание базы данных с помощью полученных запросов. В MySQL выполняется с помощью непосредственного выполнения данных запросов из файла. Файл создается с помощью кнопки “Create a new SQL”, либо открывается из готового файла.

Я не знаю че писать q-q

4. Создание приложения

## **4.1 Разработка шаблонов и макетов**

Вся графическая составляющая программы будет создана с помощью библиотек PyQt5.

Для создания шаблонов и макетов был использован вспомогательный инструмент Qt Creator, являющийся инструментом работы с объектами PyQt. Данная программа позволяет быстро и удобно составить дизайн необходимого макета, готового к подключению к основной программе.

С помощью неё были созданы следующие формы:

1. Форма авторизации — auth.ui. Содержит поля ввода логина и пароля. Содержит кнопку авторизации;
2. Форма главного меню администратора — admin.ui. Содержит кнопки перехода по таблицам Сотрудники, Заказы, Товар. Содержит окно отображения таблиц. Содержит кнопку выхода назад к окну авторизации;
3. Форма главного меню продавца — seller.ui. Содержит кнопки перехода по таблицам заказов и списка продуктов. Содержит кнопки, вызывающие окна добавления заказа и продукта в заказ. Содержит кнопку выхода назад к окну авторизации;
4. Форма добавления заказа — dialog\_add\_order.ui. Содержит поля ввода для почтового индекса, страны, города, улицы, имени клиента, кода клиента и кода сотрудника. Содержит кнопку для подтверждения добавления заказа;
5. Форма добавления продукта в заказ — add\_product.ui. Содержит поля ввода кода заказа, наименования товара и количества этого товара. Содержит кнопки добавления товара в заказ и закрытия окна.

## **4.2 Разработка интерфейсов**

Описать взаимодействие формочек

## **4.3 организация связей с БД**

Описать модуль БД и его взаимодействие с гуи

ЗАКЛЮЧЕНИЕ

В жизни таким больше не займусь добровольно

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

ПРИЛОЖЕНИЕ А. SQL запросы

## **1. Запросы на создание таблиц**

CREATE TABLE IF NOT EXISTS Client

(

ClientID INT PRIMARY KEY auto\_increment,

Name VARCHAR(100) NOT NULL,

Surname VARCHAR(100) NOT NULL,

email VARCHAR(100) NOT NULL,

phone VARCHAR(11),

country VARCHAR(100) NOT NULL,

city VARCHAR(100) NOT NULL,

street VARCHAR(100) NOT NULL,

post\_index VARCHAR(10) NOT NULL,

login VARCHAR(100) NOT NULL,

password VARCHAR(50) NOT NULL

);

CREATE TABLE IF NOT EXISTS Company

(

CompanyID INT PRIMARY KEY auto\_increment,

Name VARCHAR(100) NOT NULL,

Email VARCHAR(100),

Phone VARCHAR(11)

);

CREATE TABLE IF NOT EXISTS Employee

(

EmployeeID INT PRIMARY KEY auto\_increment,

Name VARCHAR(100) NOT NULL,

Surname VARCHAR(100) NOT NULL,

Post VARCHAR(100) NOT NULL,

Phone VARCHAR(11) NOT NULL,

email VARCHAR(100) NOT NULL,

login VARCHAR(100) NOT NULL,

password VARCHAR(50) NOT NULL

);

CREATE TABLE IF NOT EXISTS Product

(

ProductID INT PRIMARY KEY auto\_increment,

Name VARCHAR(100) NOT NULL,

Cost INT NOT NULL,

Availability INT NOT NULL,

CompanyID INT NOT NULL,

FOREIGN KEY (CompanyID) REFERENCES Company(CompanyID)

);

CREATE TABLE IF NOT EXISTS Orders

(

OrderID INT PRIMARY KEY auto\_increment,

post\_index VARCHAR(10) NOT NULL,

country VARCHAR(100) NOT NULL,

city VARCHAR(100) NOT NULL,

street VARCHAR(100) NOT NULL,

Client\_Name VARCHAR(100) NOT NULL,

ClientID INT NOT NULL,

EmployeeID INT NOT NULL,

FOREIGN KEY (ClientID) REFERENCES Client(ClientID),

FOREIGN KEY (EmployeeID) REFERENCES Employee(EmployeeID)

);

CREATE TABLE IF NOT EXISTS Shopping\_list

(

ListID INT PRIMARY KEY auto\_increment,

Price INT NOT NULL,

Amount INT NOT NULL,

OrderID INT NOT NULL,

ProductID INT NOT NULL,

FOREIGN KEY (OrderID) REFERENCES Orders(OrderID),

FOREIGN KEY (ProductID) REFERENCES Product(ProductID)

);

## **2. Запросы на добавление данных**

INSERT INTO `internetshop`.`client` (`Name`,`Surname`,`email`,`phone`,`country`,`city`,`street`,`post\_index`,`login`,`password`) VALUES ('Олег','Волков','volkov@v.ru','88005553535','Россия','г. Ростов','улица Текстильщиков, 26','152150','volk','AA76KNA4');

INSERT INTO `internetshop`.`client` (`Name`,`Surname`,`email`,`phone`,`country`,`city`,`street`,`post\_index`,`login`,`password`) VALUES ('Валентина','Трущук','Trushyk@v.ru','89255358756','Россия','г. Владимир','Ново-Ямская улица, 50','600015','osetr','jctnhjdfzbuhf');

INSERT INTO `internetshop`.`client` (`Name`,`Surname`,`email`,`phone`,`country`,`city`,`street`,`post\_index`,`login`,`password`) VALUES ('Распутин','Ефимович','ra-ra-rasputin@v.ru','9876543232','Россия','г. Красноярск','Авиационная улица, 18','660020','rasputin','tar-tar-taglia');

INSERT INTO `internetshop`.`employee` (`Name`,`Surname`,`Post`,`Phone`,`email`,`login`,`password') VALUES('Константин','Казах','Администратор','87655679191','kazuha@v.ru','kazakh','kazuzuzuzu');

INSERT INTO `internetshop`.`employee` (`Name`,`Surname`,`Post`,`Phone`,`email`,`login`,`password`) VALUES('Раиса','Сенная','Продавец','87652345382','raiden@v.ru','raiden','rararaidededen');

INSERT INTO `internetshop`.`employee` (`Name`,`Surname`,`Post`,`Phone`,`email`,`login`,`password`) VALUES('Андрей','Камин','Продавец','84985126784','ayato@v.ru','ayato','kamisato');

INSERT INTO `internetshop`.`company` (`Name`, `Email`, `Phone`) VALUES ('Favonius', 'favonius@v.ru', '87459812323');

INSERT INTO `internetshop`.`company` (`Name`, `Email`, `Phone`) VALUES ('Fatui', 'fatui@v.ru', '89348365959');

INSERT INTO `internetshop`.`company` (`Name`, `Email`, `Phone`) VALUES ('Qixing', 'qixing@v.ru', '89123458282');

INSERT INTO `internetshop`.`product`(`Name`,`Cost`,`Availability`,`CompanyID`) VALUES('Сыр','100','1200','1');

INSERT INTO `internetshop`.`product`(`Name`,`Cost`,`Availability`,`CompanyID`) VALUES('Молоко','100','1200','1');

INSERT INTO `internetshop`.`product`(`Name`,`Cost`,`Availability`,`CompanyID`) VALUES('Капуста','100','1200','1');

INSERT INTO `internetshop`.`product`(`Name`,`Cost`,`Availability`,`CompanyID`) VALUES('Огненная вода','1090','1200','2');

INSERT INTO `internetshop`.`product`(`Name`,`Cost`,`Availability`,`CompanyID`) VALUES('Пельмени','200','1200','2');

INSERT INTO `internetshop`.`product`(`Name`,`Cost`,`Availability`,`CompanyID`) VALUES('Красная икра','900','1200','2');

INSERT INTO `internetshop`.`product`(`Name`,`Cost`,`Availability`,`CompanyID`) VALUES('Креветки','500','1200','3');

INSERT INTO `internetshop`.`product`(`Name`,`Cost`,`Availability`,`CompanyID`) VALUES('Семена лотоса','500','1200','3');

INSERT INTO `internetshop`.`product`(`Name`,`Cost`,`Availability`,`CompanyID`) VALUES('Перец чили','250','1200','3');