Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

Пермский национальный исследовательский политехнический университет

Электротехнический факультет

Кафедра Информационные технологии и автоматизированные системы

Курсовая работа по дисциплине

«Базы данных»

Вариант 73

Тема: «Разработка базы данных “Система управления аптекой”»

Выполнил: студент группы РИС-20-1Б

Эскарус Б. П.

Проверил: Доцент кафедры ИТАС

к.т.н. Петренко А. А.

г. Пермь – 2022

**Оглавление**

[Реферат...........................................................................................................3](#_gjdgxs)

[Введение.........................................................................................................4](#_30j0zll)

[1. Анализ предметной области.....................................................................6](#_1fob9te)

2. Проектирование базы данных............................................................... 11

2.1 Построение концептуальной модели базы данных ..............12

2.2 Построение логической модели базы данных .......................14

2.3 Построение физической модели базы данных.......................16

2.4 Выбор СУБД .............................................................................18

2.4.1Сравнение различных СУБД.........................................18

3.Технология реализации программы.......................................................20

3.1 Реализация функций и интерфейса ............................................20

3.2 Разграничение прав доступа........................................................32

4.Заключение ..............................................................................................33

5.Список литературы..................................................................................34

**Реферат**

Общий объем: (35) стр. Количество иллюстраций: (32).

Число использованных источников информации: (7).

Ключевые слова: Реляционная база данных, база данных, модели баз данных, Концептуальная модель, Логическая модель, Физическая модель, таблица, СУБД.

Объект исследования: информационная система управления аптекой.

Предмет исследования: информационная система для хранения данных об операциях, которые происходят в аптеке.

В ходе работы была реализована база данных для хранения информации об информационной системе ВУЗа, а также пользовательский интерфейс для работы с БД.

Методы исследования: Анализ, сравнение, эксперимент,

моделирование.

**Введение**

В наше время большинство организаций, в том числе медицинские, переходят на цифровой способ хранения и обработки информации, поэтому востребованность баз данных(БД) заметно увеличилась. Без БД сейчас не обходится ни одна крупная корпорация, каждое крупнейшее предприятие хранит определенную информацию в собственной базе данных.

В современных базах данных есть возможность хранить огромный набор информации, начиная от данных в самом простом виде, заканчивая сложными логически-связанными структурами.

Сейчас, зачастую, большую часть логически-связанной информации хранят в реляционных базах данных. В них, в отличие от нереляционных баз данных, используется табличная система строк и столбцов, а также связей между таблицами. Такой подход позволяет удобно структурировать, хранить и редактировать информацию.

Постановка задачи: Работа начинается в аптеке, которая делится на комбинированные препараты и готовые препараты, изготовленные на заводе.

Лекарства, произведенные только на заводах. Данные о покупке, продаже и хранении клиентов, которые покупают лекарства. клиент, который запрашивает счет-фактуру с полным именем, адресом, номером телефона и типом клиента, является ли он постоянным клиентом или клиентом на один раз.

Что касается лекарственных средств, которые устанавливаются внутри аптеки, то они также подлежат купле-продаже, реквизитам счета и данным о клиентах, но к этому есть особое дополнение, из чего состоят эти лекарства с точки зрения материалов, участвующих в их изготовлении. производство, их количество, цены на эти материалы, а также годность этих материалов и их количество, когда они были куплены и сколько осталось в аптеке этих веществ, так что есть разница между готовыми лекарствами изготовленные на заводе, и составные лекарства внутри аптеки, поэтому между ними есть разница, но, в конце концов, между ними есть общие характеристики, которые представляют собой процессы покупки и продажи, но разница между ними заключается в компонентах соединения. лекарства и способы их приготовления в аптеке.

Поэтому создается база данных, которая организует все эти данные и соединяет эту базу данных с интерфейсом, облегчающим работу пользователя с базой данных.

Задачи:

1. Провести анализ предметной области.

2.Выбрать подходящее программное обеспечение для реализации программы

3.Разработать концептуальную, логическую и физическую модели данных

4.Разработать базу данных

5.Разработать программу для работы с базой данных

Методы исследования: Анализ, сравнение, эксперимент.

В ходе работы была реализована база данных для хранения информации об операциях, которые происходят в аптеке, а также пользовательский интерфейс для работы с БД

**1. Анализ предметной области**

База данных — это упорядоченный набор структурированной информации или данных, которые хранятся в электронном виде в компьютерной системе.

Управление БД производится с помощью систем управления базами данных(СУБД). СУБД представляет собой совокупность некоторых программных средств, которые обеспечивают управление, создание и использование базы данных.

Реляционные базы данных – это набор данных(таблиц) со связями между ними. Таблицыразделены на строки и столбцы, на пересечении которых содержатся значения данных. Каждая таблица имеет уникальное имя*,* описывающее ее содер­жимое. Каждая строка таблицы представляет собой набор связанных значений а столбец хранит в себе определенный тип данных.

Разработка пользовательского интерфейса(программы) для работы с данными потребует использование интегрированной среды разработки, а также структуру, которая станет основой интерфейса.

Для разработки базы данных необходимо использовать СУБД, которая предоставит все необходимые возможности для реализации требований к программе.

Система управления аптекой представляет собой набор данных, в котором хранятся лекарственные средства и количества в магазине изготовленных лекарственных средств, составных лекарственных средств и сырья, используемого при изготовлении составных лекарственных средств.Готовые лекарственные средства или закупки сырья, которые используются в составе комбинированные лекарства.

Общее правило состоит из синтезированных лекарств, например, из названия лекарства и его эффективности и цены продажи покупателю и даты изготовления и срока годности и вида какого-либо смысла, который он следует к какой-либо группе, делаете ли вы таблетки или жидкость и другие виды лекарств или составные лекарства в аптеке они имеют особый характер качеств они след хамия существующие качества лекарств выпускаемых на фабриках в дополнение к составу любого лекарства другими словами из чего состоит это лекарство метод состава и цены на композиционные материалы в т.ч. и другие свойства, о которых будет подробно рассказано.

Информационная система должна позволять получать ответы на запросы:

1) Получите сведения о покупателях, которые не пришли забрать свой заказ в назначенное им время, и общее число покупателей.

2) Получите перечень и общее число покупателей, которые ждут прибытия на склад нужных им медикаментов в целом и по указанной категории медикаментов.

3) Получите перечень десяти наиболее часто используемых медикаментов в целом и в указанной категории медикаментов.

4) Подсчитайте, какой объем указанных веществ использован за указанный период.

5) Получите перечень и общее число покупателей, заказывавших определенное лекарство или определенные типы лекарств за данный период.

6) Получите перечень и типы лекарств, достигших своей критической нормы или закончившихся.

7) Получите перечень лекарств с минимальным запасом на складе в целом и по указанной категории медикаментов.

8) Получите полный перечень и общее число заказов, находящихся в производстве.

9) Получите полный перечень и общее число препаратов, требующихся заказам, находящимся в производстве.

10) Получите все технологии приготовления лекарств указанных типов, конкретных лекарств, лекарств, находящихся в справочнике заказов в производстве.

11) Получите сведения о ценах на указанное лекарство в готовом виде, об объеме и ценах на все компоненты, требующиеся для этого лекарства.

12) Получите сведения о постоянных клиентах медикаментов.

13) Получите сведения о конкретном лекарстве (его тип, способ приготовления, названия всех компонент, цены, его количество на складе).

**1.1 Требования к программе:**

1. В базе данных должны храниться сведения о покупателях, готовых лекарствах, которые были изготовлены на заводах, комбинированных лекарствах, которые были установлены в аптеке, составе этих лекарств, способах изготовления, видах лекарств и типах покупателей.
2. База данных должна содержать информацию о лекарственных средствах с даты изготовления и срока годности.
3. База данных должна содержать информацию о клиентах, такую как полное имя, адрес клиента, номер телефона, тип клиента, является ли он постоянным клиентом или разовым клиентом.
4. База данных должна содержать информацию о заказе от владельца заказа и о том, кто ему продал заказ, дату продажи, стоимость инвойса, величину скидки, полученной клиентом и какие лекарства клиент приобрел.
5. Программа должна позволять пользователю просматривать все таблицы и данные содержащиеся в них.
6. Программа должна позволять пользователю добавлять данные в таблицы базы данных.
7. Программа должна позволять пользователю редактировать данные в таблицах базы данных.
8. Программа должна позволять пользователю удалять данные из таблиц базы данных.
9. Система должна позволять пользователю реализовать запрос на получение списка лекарственных средств определенных категорий и видов по типу лекарства или по типу производителя этого лекарства.
10. Система должна позволять пользователю реализовать запрос списка хранимого сырья.
11. Система должна позволять пользователю заполнить запрос на методы приготовления лекарств.
12. Система должна позволять пользователю выполнить запрос на получение ингредиентов всех лекарств в аптеке.
13. Система должна позволять пользователю выполнять запрос информации о Uders, который все еще находится в стадии подготовки.
14. Система должна позволять пользователю выполнять запрос информации о клиентах, которые не получили свои заказы.
15. Система должна позволять пользователю выполнять запрос информации о покупателях, которые не получили свои заявки, поскольку она недоступна в аптеке.
16. Система должна позволять пользователю выполнить запрос на получение информации о клиентах, которые не получили их применения из-за отсутствия доступности сырья, хранящихся внутри кризиса для установки этих препаратов.
17. Система должна позволять пользователю выполнять запрос на получение информации о лекарственных средствах, которые вблизи или уже реализованы.
18. Система должна позволять пользователю выполнять запрос информации о сырье, которое заканчивается или уже заканчивается.
19. Система должна позволять пользователю осуществлять запрос информации о постоянных и непостоянных клиентах.
20. Система должна позволять пользователю реализовать запрос на получение информации о препаратах определенной категории, по их компонентам, способам изготовления и ценам на каждый компонент, входящий в их состав.
21. Система должна позволять пользователю выполнять запрос информации о сырье, хранящемся на складе.

Проектирование базы данных — один из важнейших процессов при разработке БД. Процесс представляет собой создание схемы базы данных и определения необходимых ограничений целостности.

Основные задачи проектирования: Обеспечение хранения в БД всей необходимой информации, обеспечение возможности получения данных по всем необходимым запросам, предусмотрение избежания дублирования данных, обеспечение целостности базы данных.

При проектировании баз данных на раннем этапе создаются 3 модели данных: концептуальная, логическая и физическая.

Для разработки системы, следующей за Веб-Аблкишн, было одобрено использование языков программирования (HTML,CSS,PHP), которые облегчают работу над созданием системы Аблкишн-Веб-специальной аптеки.

**Вывод**

Конечным продуктом должна быть база данных в дополнение к веб-приложению, упрощающему работу пользователя с базой данных.

База данных должна предоставлять возмозможность хранить и обрабатывать необходимые данные. Приложение должно предоставлять возможность пользователю взаимодействовать с базой данных и данными, хранящимися в ней.

**2. Проектирование базы данных**

Необходимые для разработки базы данных модели, будут созданы с помощью инструмента Draw.IO, так как он является наиболее удобным среди прочих и отвечает всем требованиям, которые необходимы для моделирования БД.

Для единообразия программирования баз данных введены следующие понятия для моделей данных:

1. Объект или сущность. Сущность – класс однотипных объектов, информация о которых имеет существенное значение для рассматриваемой предметной области.
2. Атрибут. Атрибут - это характеристика объекта, соответствующая его сущности.
3. Третье понятие в проектировании концептуальной базы данных это связь или отношения между объектами.

## **2.1 Построение концептуальной модели базы данных**

Концептуальная модель [базы данных](https://webonto.ru/chto-takoe-baza-dannyih/) это некая наглядная диаграмма, нарисованная в принятых обозначениях и подробно показывающая связь между объектами и их характеристиками.

Концептуальная модель была построена по нотации Питера Чена.

Ход работы:

1. Создание таблиц, которые в полной мере отразят необходимую информацию.
2. Создание связей между таблицами, которые позволят пользователю корректно взаимодействовать с нужными данными.

Концептуальная модель представлена на Рисунке 1.1.

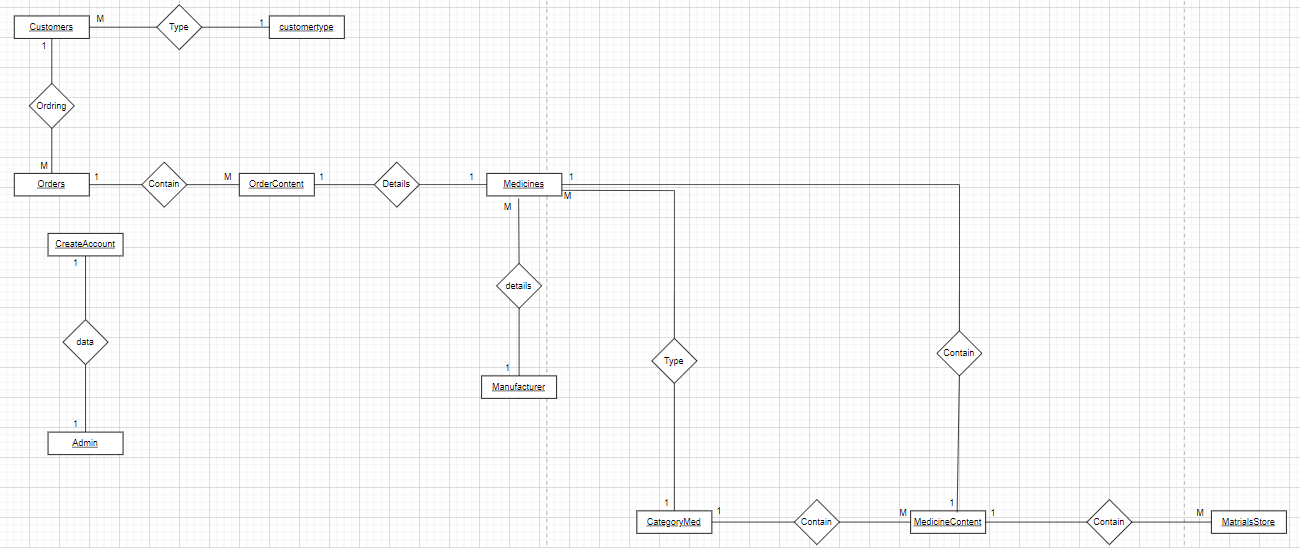


Рисунок 1.1 - Концептуальная модель базы данных

Разработанная модель базы данных содержит 11 сущностей и 13 связь, которые позволят вполне корректно отображать и взаимодействовать с информацией базы данных “Система управления аптекой”.

Таблицы:

1. ‘customers’ имя клиента все сохраненные данные и номер телефона и адрес клиента и тип сделки клиента с аптекой.
2. ‘cutomertype’ была постоянной или в одном клиенте хранится типы клиентов, нужно ли на этот раз добиться условий нормализации и предотвратить повторение.
3. ‘Orders’ данные хранятся в основном приложении, такие как дата стоимости и скидки и спрос.
4. ‘ordercontent’ Лекарства, запрошенные в этом приложении, их названия и цены сохраняются.
5. ‘medicines’ сохранения в этой таблице всех данные от имени медицины и эффективность цены и дата действия этого лекарства и исходного продукта для этой медицины.
6. ‘manufacturer’ это таблица, в которой хранится фабрика, которая является фабрикой или аптекой, и это для предотвращения повторения и достижения условия нормализации.
7. ‘categorymed’ хранит категории лекарств, помогая предотвратить повторение.
8. ‘medicinecontent’ хранит компоненты лекарственного средства через ключи, чтобы предотвратить повторение Dion, а также хранить способ изготовления лекарства и количество каждого сырья в этой медицине.
9. ‘materialsstore’ это аптечный магазин, в котором хранится сырье, материалы и все данные, начиная с названия артикула и цены за каждые сто грамм и заканчивая сроком годности, производителем и сроком годности.
10. ‘Register’ хранит данные системного пользователя при первом входе в систему
11. ‘Logindata’ хранит адрес электронной почты и пароль пользователя, чтобы он мог каждый раз входить в систему.

**2.2 Построение логической модели базы данных**

Логическая модель данных является начальным прототипом будущей базы данных. Логическая модель строится в терминах информационных единиц, но без привязки к конкретной СУБД.

Во время разработки логической модели данных были определены такие понятия, как первичный ключ - в реляционной модели данных один из потенциальных ключей отношения, выбранный в качестве основного ключа, внешний ключ - столбец или комбинация столбцов, значения которых соответствуют Первичному ключу в другой таблице

Логическая модель была построена по нотации Мартина.

Ход работы:

1. Определение первичных и внешних ключей для каждой сущности базы данных.
2. Построение связей непосредственно между соответствующими ключами сущностей.
3. Было принято решение, что в качестве первичного ключа каждой таблицы будет взят ID записи.

Логическая модель изображена на Рисунке 1.2.

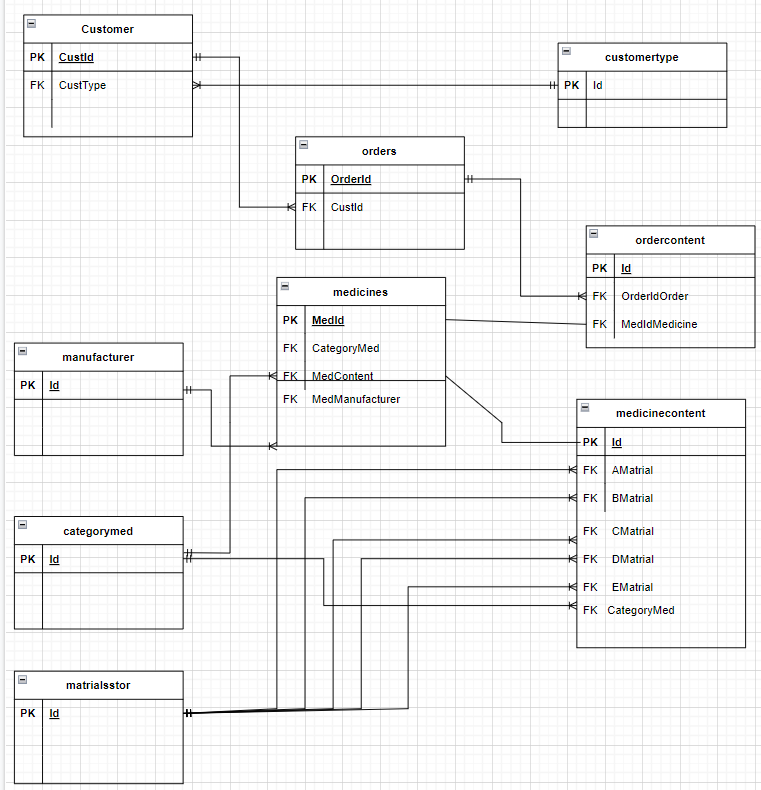


Рисунок 1.2 - Логическая модель базы данных.

Таблицы:

1. ‘customers’ Первичный ключ “Custid”. Внешние ключи “CustType”.
2. ‘cutomertype’ Первичный ключ “Id”.
3. ‘Orders’ Первичный ключ “OrderId”. Внешние ключи “CustId”.
4. ‘ordercontent’ Первичный ключ “Id”. Внешние ключи “OrderIdOrder и MedIdMedicine”.
5. ‘medicines’ Первичный ключ “MedId”. Внешние ключи “CategoryMed , MedContent и MedManufacturer”.
6. ‘manufacturer’ Первичный ключ “Id”.
7. ‘categorymed’ Первичный ключ “Id”.
8. ‘medicinecontent’ Первичный ключ “Id”. Внешние ключи “AMatrial ,BMatrial , CMatrial ,DMatrial,EMatrial и CategoryMed”.
9. ‘materialsstore’ Первичный ключ “Id”.
10. ‘Register’ Первичный ключ “RegId”.
11. ‘Logindata’ Первичный ключ “LogId”. Внешние ключи “Reg”.

**2.3 Построение физической модели базы данных**

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

Ход работы:

1. Добавление неключевых полей в таблицы.
2. Определение типов данных для каждого поля.
3. Определение авто-инкрементных и ненулевых полей.
4. Определение дефолтных значений и проверок CHECK в определенных сущностях

Во время разработки физической модели данных было принято решение использовать следующие типы данных:

1. INT – целочисленный тип данных.
2. VARCHAR(?) - Строковый тип, может содержать цифры, латиницу, кириллицу и спецсимволы. Длина строки <=?.
3. VARCHAR(1) - Строковый тип, длина строки = 1, используется для получения бинарного значения.
4. DATE - тип данных содержащий информацию о дате в формате “ГГГГ-ММ-ДД”

Физическая модель была построена по нотации Мартина.

Физическая модель изображена на Рисунке 1.3.

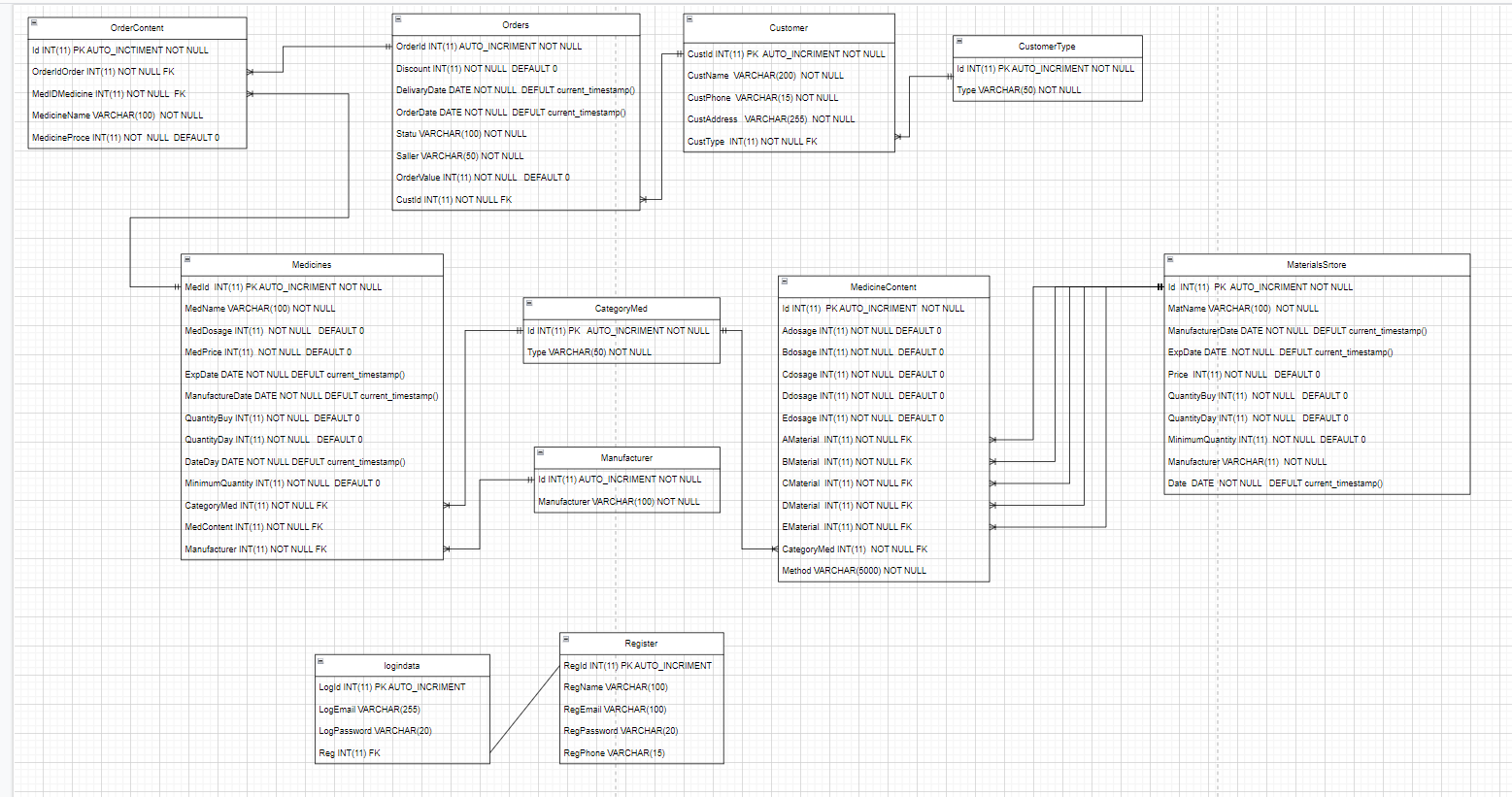


Рисунок 1.3 - Физическая модель базы данных

Разработанная модель содержит достаточное для корректной работы приложения количество ограничений CHECK, определенных значений DEFAULT. Значения бинарных полей определены в примечаниям к таблице.

**2.4 Выбор СУБД**

На основании логической модели данных и требований к программе был проведен выбор СУБД для реализации БД.

Для того чтобы выбрать наиболее подходящую СУБД был проведен сравнительный анализ двух СУБД .

**2.4.1Сравнение различных СУБД**

**MySQL:**

Наличие справочного материала в свободном доступе:

1. Большое количество справочного материала для работы с СУБД.

Интерфейс работы с СУБД:

1. Работа с БД осуществляется с помощью веб-интерфейса PhpMyAdmin, что позволяет работать без установки лишнего программного обеспечения.

Возможность подключения к выбранной среде разработки:

1. Свободное подключение СУБД к выбранной среде разработки.

**MS SQL Server:**

Наличие справочного материала в свободном доступе:

1. Небольшое количество справочного материала для работы с СУБД.

Интерфейс работы с СУБД:

1. Работа с БД осуществляется с помощью ПО SQL Server Management Studio, что приводит к необходимости устанавливать дополнительное ПО.

Возможность подключения к выбранной среде разработки:

1. Подключение СУБД к выбранной среде разработки требует изучение дополнительной информации и установки дополнительного ПО.

Исходя из представленных и проанализированных данных было принято решение использовать для работы СУБД MySQL, так как данный инструмент является наиболее простым в понимании и использовании.

**Вывод**

База данных будет реализована с помощью СУБД MySql. Для отражения всех необходимых данных и запросов будет создано 11 таблиц и 13 связь.

**3. Технология реализации программы**

## **3.1 Реализация функций и интерфейса**

1. При работе с этой системой должны быть сохранены такие данные, как электронная почта и пароль, чтобы разрешить доступ к сайту, изображена на Рисунке 1.4.

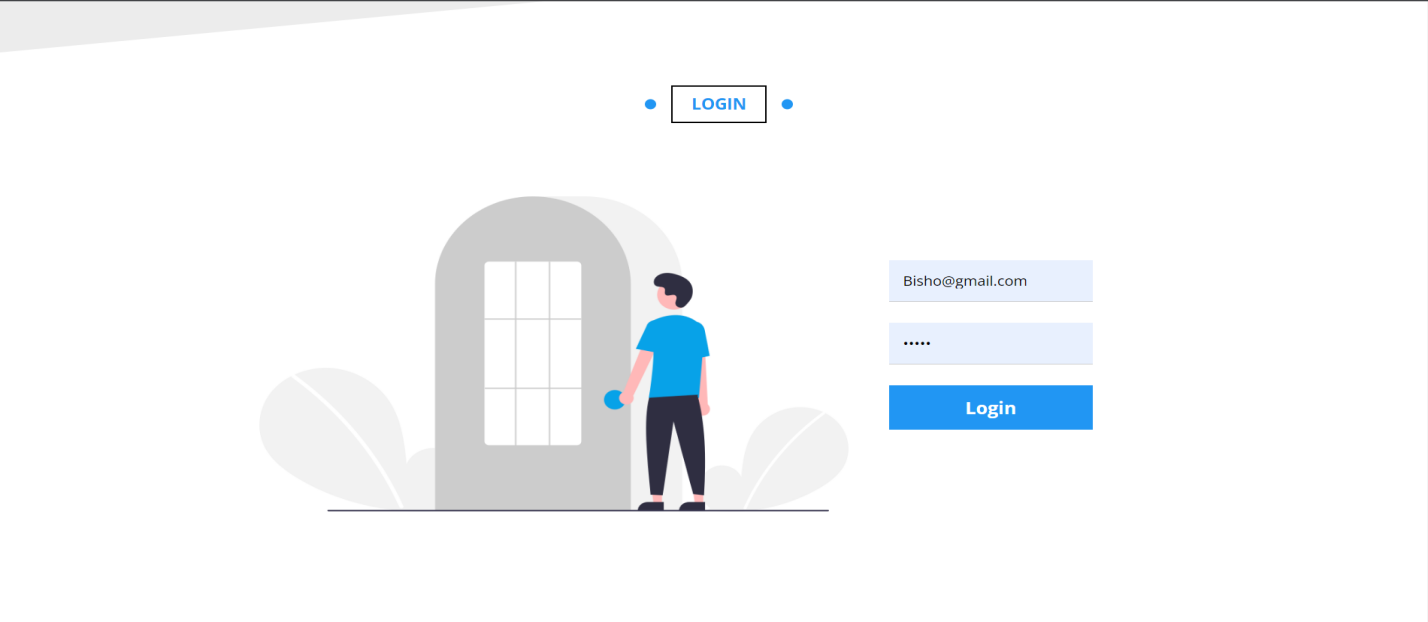


Рисунок 1.4 - Интерфейс входа на сайт.

2. Для того, чтобы пользователь мог войти, данные должны быть записаны в том случае, если пользователь входит на сайт в первый раз,изображена на Рисунке 1.5.

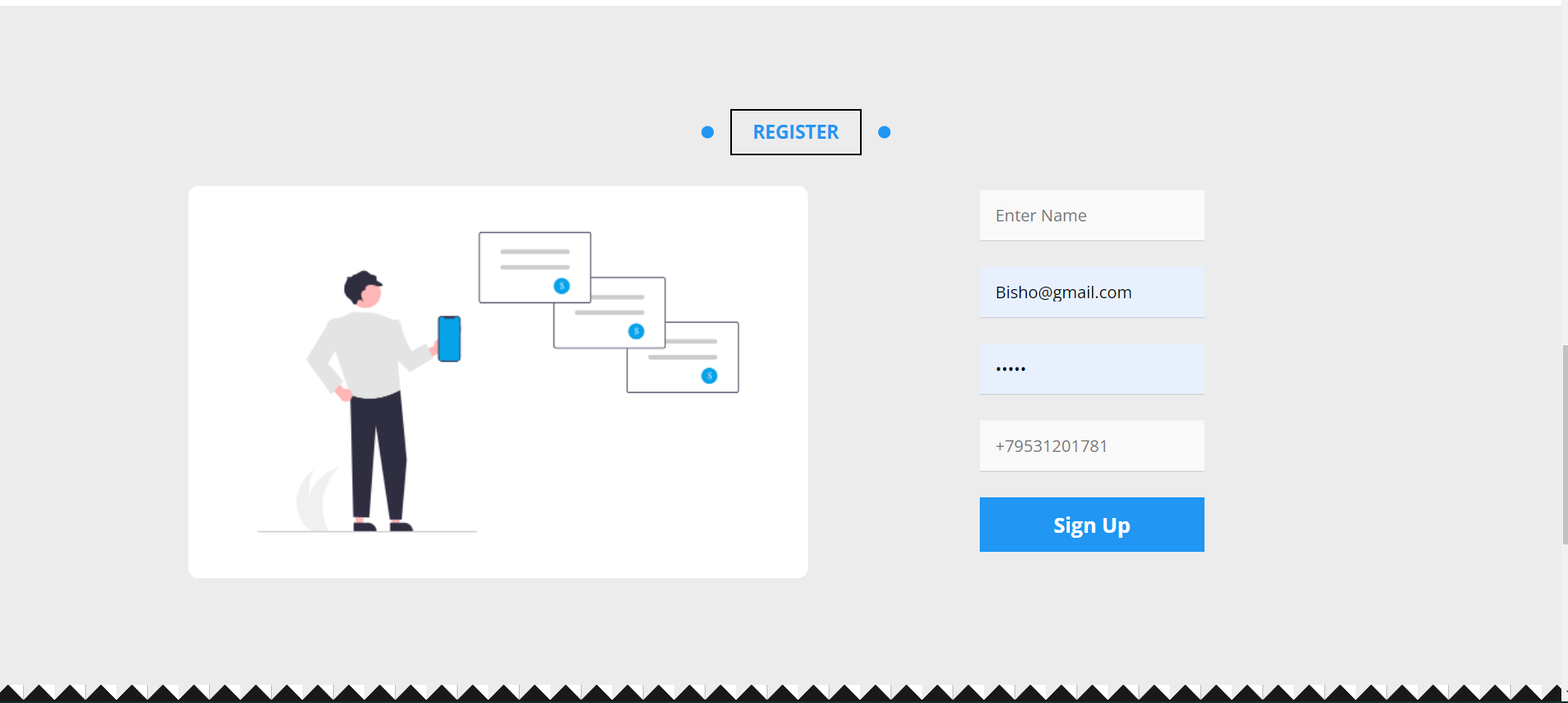


Рисунок 1.5 -Интерфейс для создания электронной почты и пароля.

3. Из этих двух таблиц понятно, какие данные требуются для хранения и связь между ними и как связать эти две таблицы связью один к одному,изображена на Рисунке 1.6.

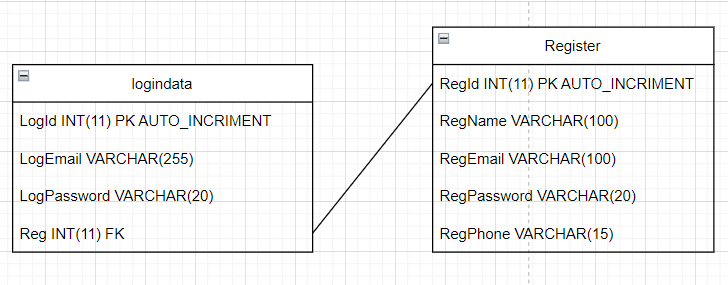


Рисунок 1.6 - Таблицы хранения пользовательских данных.

4.Вот панель управления для всего в системе,изображена на Рисунке 1.7.

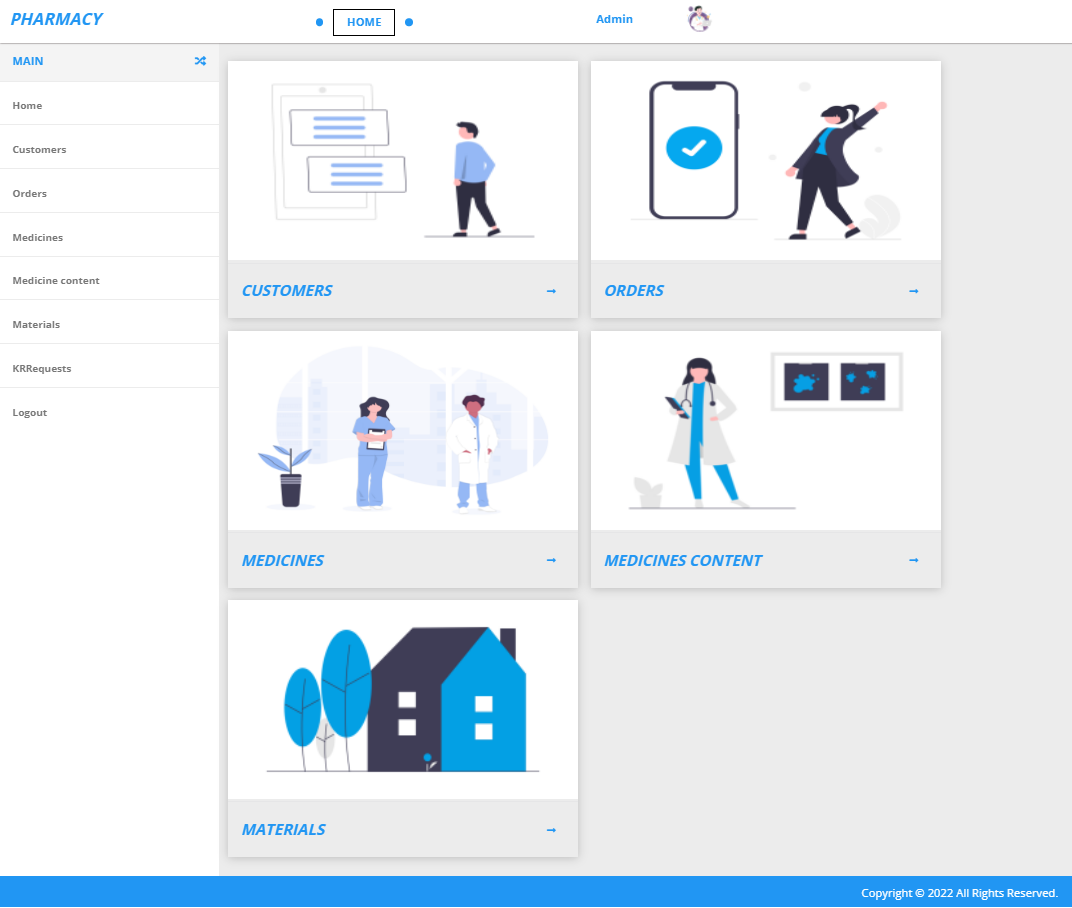


Рисунок 1.7 - панель управления.

1. Например, есть процесс добавления клиента после ввода всех необходимых данных,изображена на Рисунке 1.8.

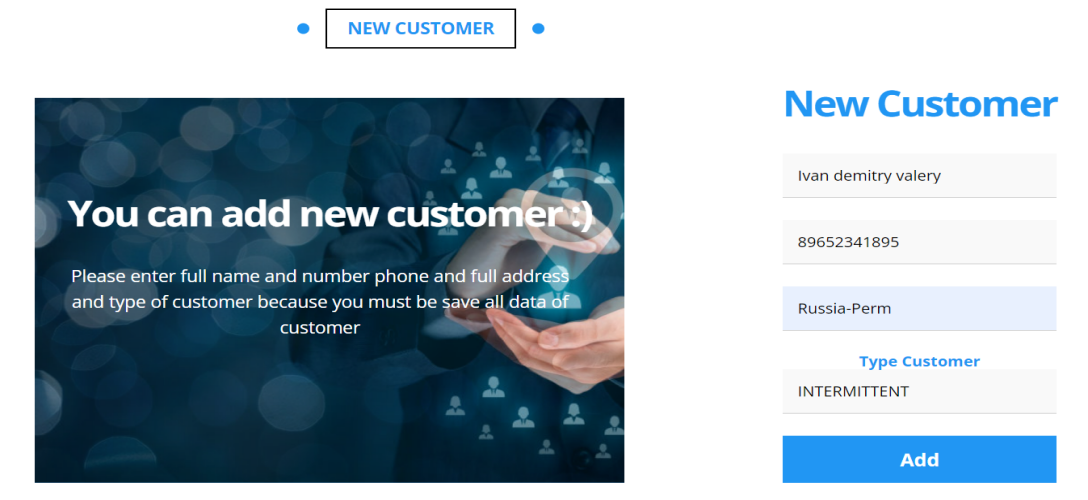


Рисунок 1.8

Было использовано добавление запрос, убедившись, что эти данные необходимы,изображена на Рисунке 1.9.

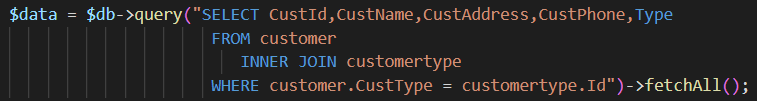
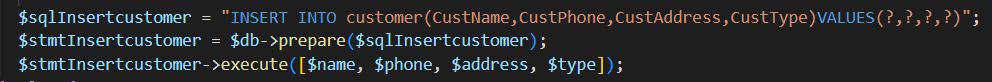


Рисунок 1.9

Затем все сохраненные данные показывают клиентов в дополнение к новому клиенту, который был добавлен,изображена на Рисунке 2.1.

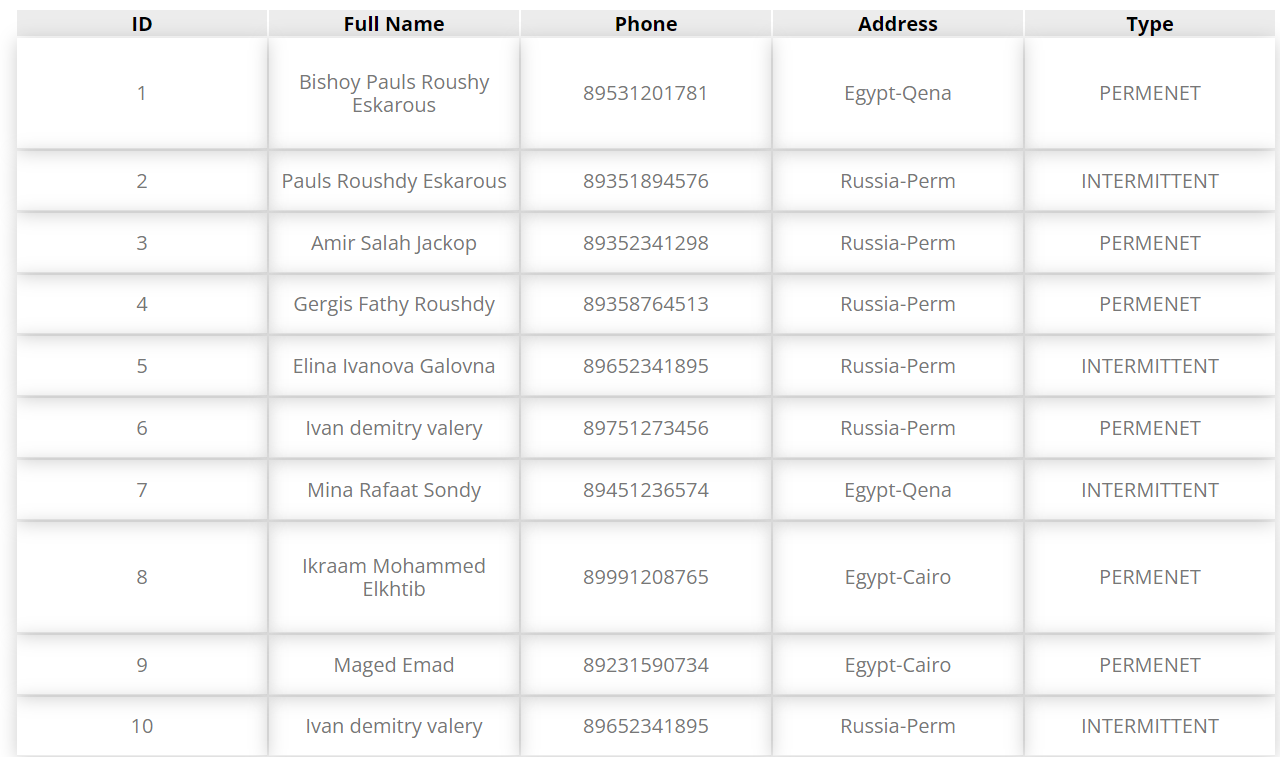


Рисунок 2.1

7. Например здесь при удалении препарата из типа таблеток отображаются все препараты этого типа,изображена на Рисунке 2.2.



Рисунок 2.2

Лекарство называется бруфен и имеет идентификационный номер 12,изображена на Рисунке 2.3.

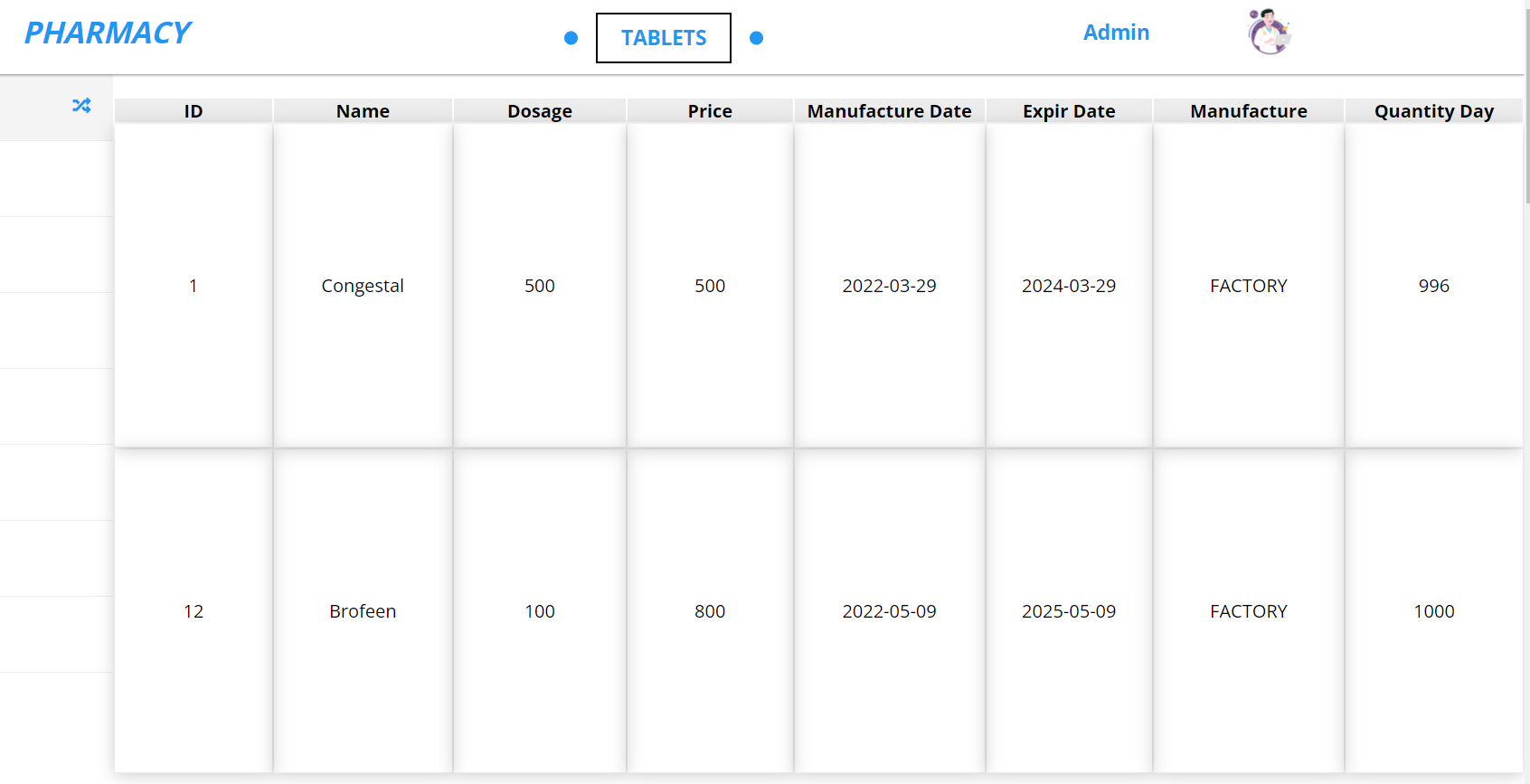


Рисунок 2.3

Удаление с помощью продюсера был создан, чтобы удалить лекарства взять одну переменную, которая идентификатор медицины требуется, чтобыудалить его,изображена на Рисунке 2.4.

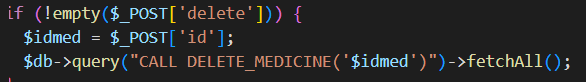


Рисунок 2.4

После ввода данных препарат необходимо удалить и отправить в базу,изображена на Рисунке 2.5.

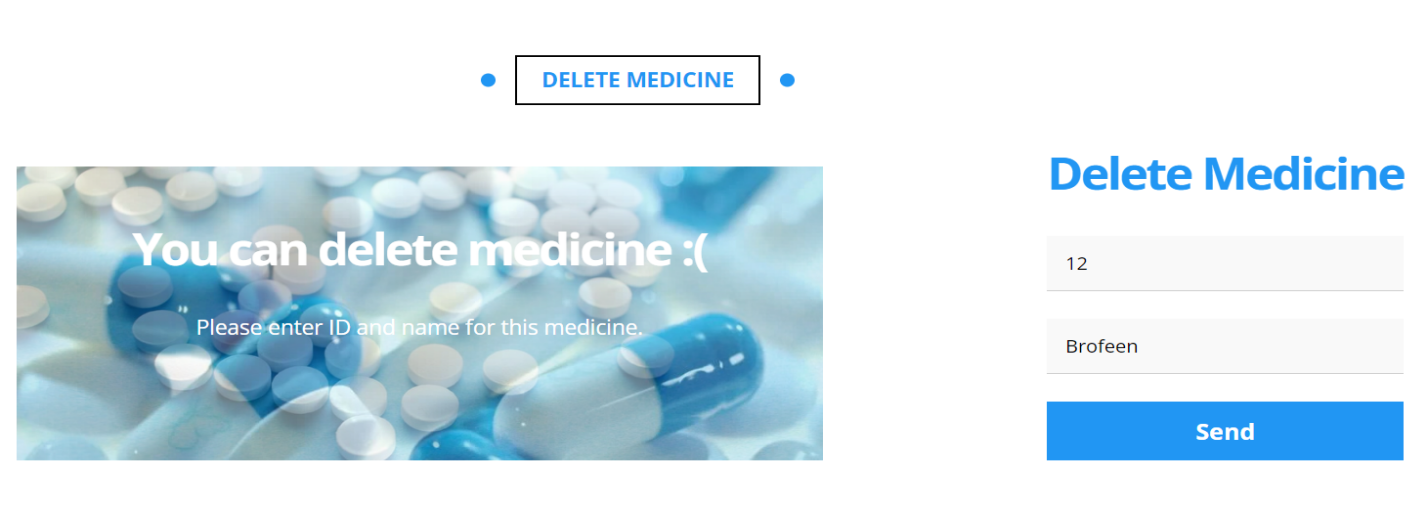
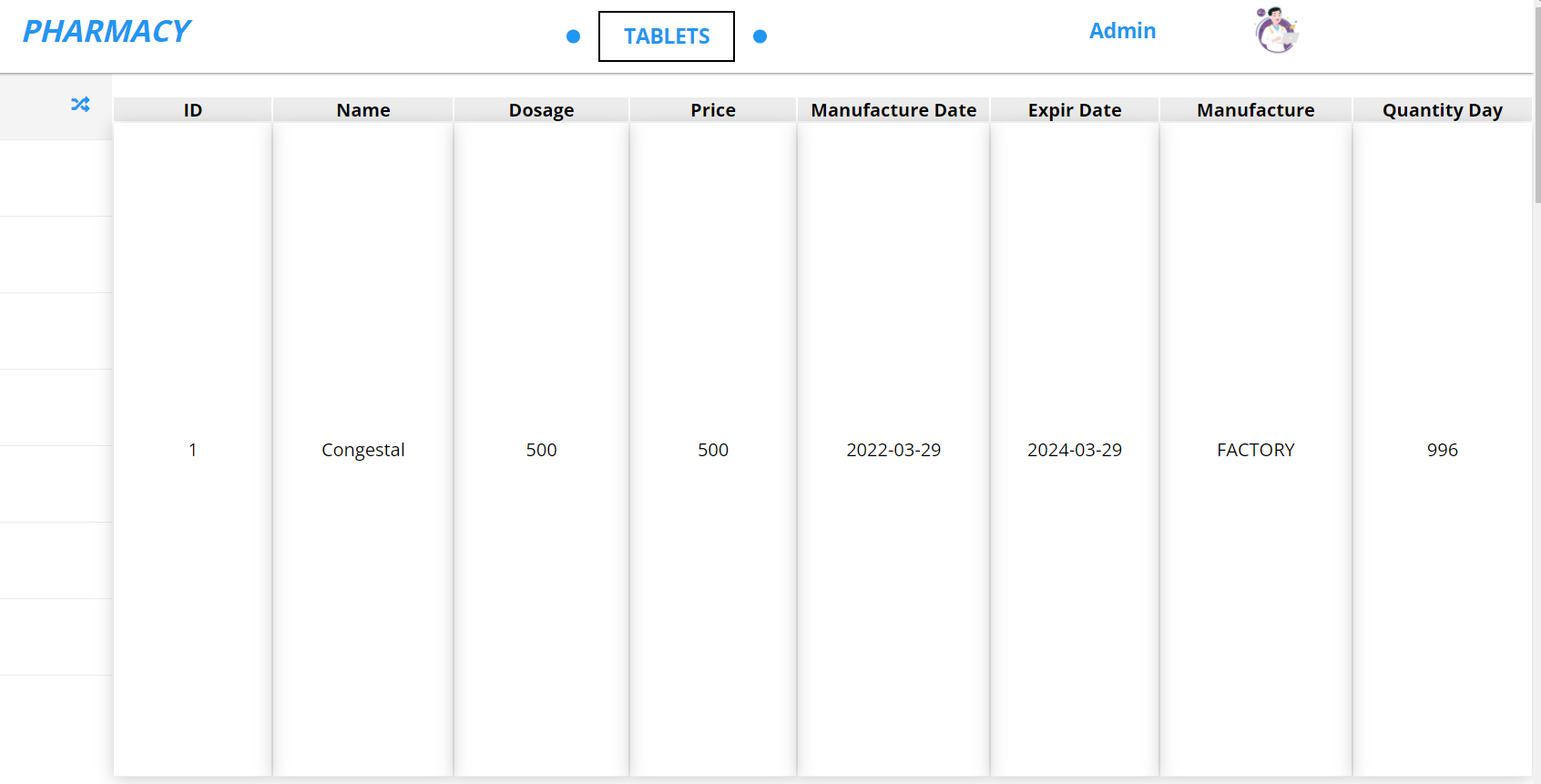


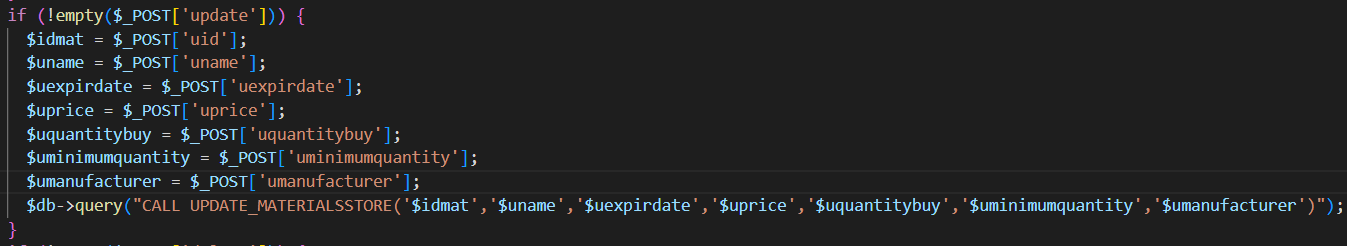
Рисунок 2.5

После этого препарат удаляется и все его данные удаляются из базы данных,изображена на Рисунке 2.6.



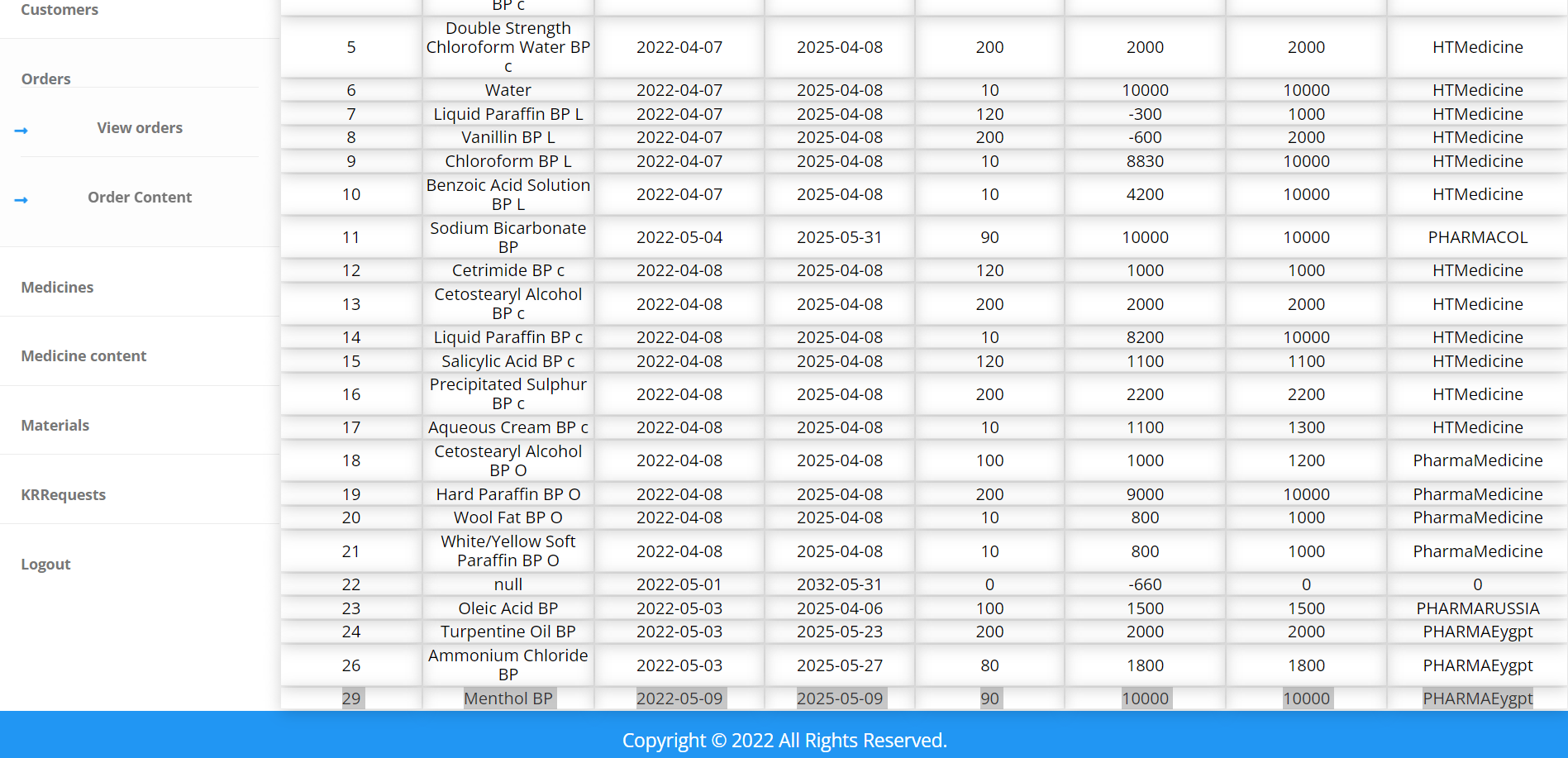
Рисунке 2.6.

1. Например, процесс изменения данных хранимого сырья.Процесс модификации данных берет данные от производителя, а пользователь изменяет базу данных,изображена на Рисунке 2.7.



Рисунке 2.7.

1. Буду изменено данные по подшипниковому идентификатору 29 будет скорректирована цена одного грамма процентов из них и суммы покупки и количества существующих в настоящее время,изображена на Рисунке 2.8.



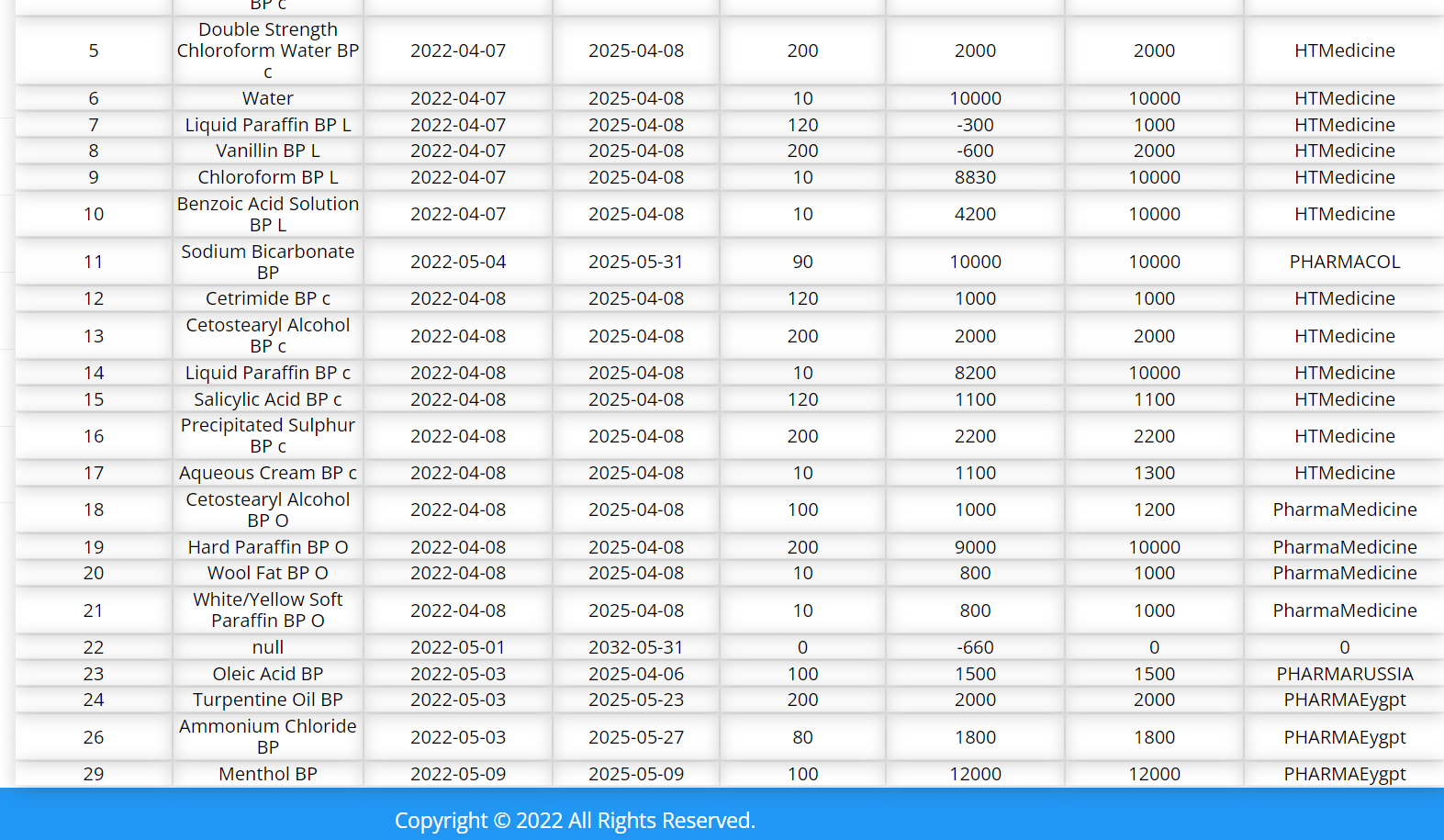
Рисунке 2.8.

Новые данные отправляются в базу данных,изображена на Рисунке 2.9.



Рисунке 2.9.

И это уже изменило цену одного грамма процентов от этого количества купленного товара и оставшегося количества,изображена на Рисунке 3.1.



Рисунке 3.1.

Былые примеры реальности самой работы по добавлениям, удаления,модификации и отображения данных будут показано, как все требования реализации требуется в этой работе.

**3.2 Запросы в информационной системе:**

1. Получите сведения о покупателях, которые не пришли забрать свой заказ в назначенное им время, и общее число покупателей,изображена на Рисунке 3.2.

CREATE VIEW KR\_1\_1

AS

SELECT CustName,CustPhone,CustAddress,DelivaryDate,OrderDate,Statu

FROM customer

INNER JOIN orders ON customer.CustId = orders.CustId

WHERE orders.Statu = 'NO RECEIVEDCUST' AND DATEDIFF(orders.DelivaryDate,NOW()) <=0

GROUP BY CustName;

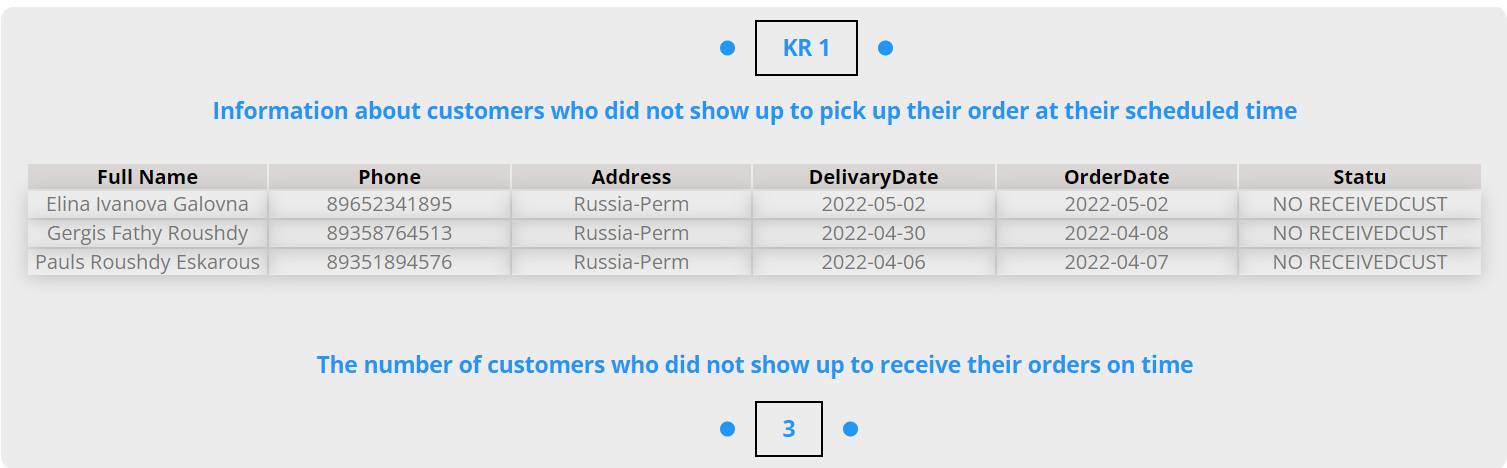
CREATE VIEW kr\_1\_2

AS

SELECT COUNT(\*) FROM customer

INNER JOIN orders ON orders.CustId = customer.CustId

WHERE orders.Statu = 'NO RECEIVEDCUST' AND DATEDIFF(orders.DelivaryDate,NOW()) <=0



Рисунке 3.2.

1. Получите перечень и общее число покупателей, которые ждут прибытия на склад нужных им медикаментов в целом и по указанной категории медикаментов,изображена на Рисунке 3.3.

CREATE VIEW KR\_2

AS

SELECT CustId,CustName,CustPhone, CustAddress,DelivaryDate, OrderDate,COUNT(OrderId) AS NumberOrders, Statu

FROM customer

INNER JOIN orders ON customer.CustId = orders.CustId

WHERE orders.Statu LIKE 'NO RECEIVEDPHAR%'

GROUP BY CustName;

DELIMITER //

CREATE PROCEDURE KR\_2\_2(IN NameCategory VARCHAR(100))

BEGIN

SELECT CustId, CustName,CustPhone, CustAddress, DelivaryDate,OrderDate,COUNT(orders.OrderId) AS NumberOrders,Statu

FROM customer

INNER JOIN orders ON customer.CustId = orders.CustIdOrder

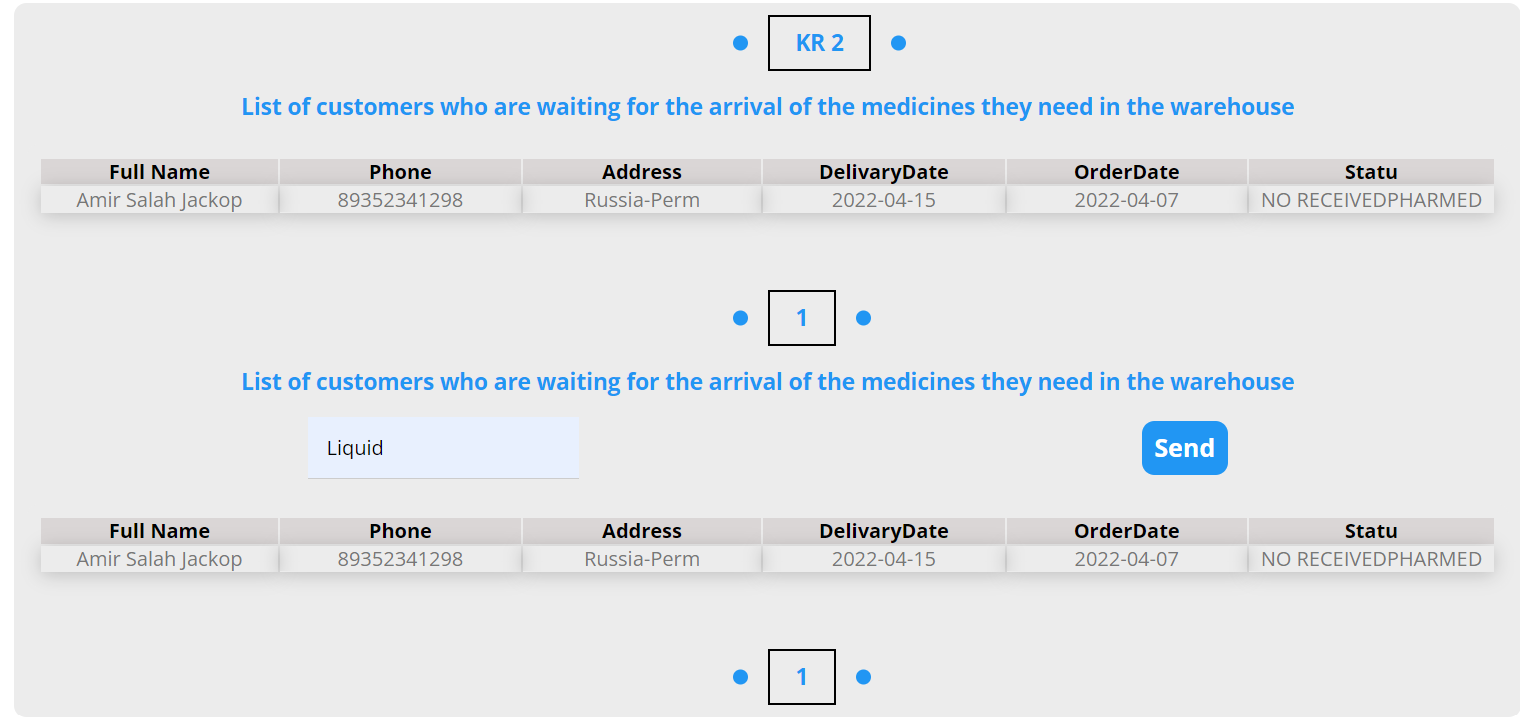
INNER JOIN categorymed ON categorymed.Id = medicines.CategoryMed

WHERE orders.Statu LIKE 'NO RECEIVEDPHARMED' AND ordercontent.MedIdMedicine = medicines.MedId AND medicines.CategoryMed = categorymed.Id AND categorymed.Type = NameCategory

GROUP BY CustName;

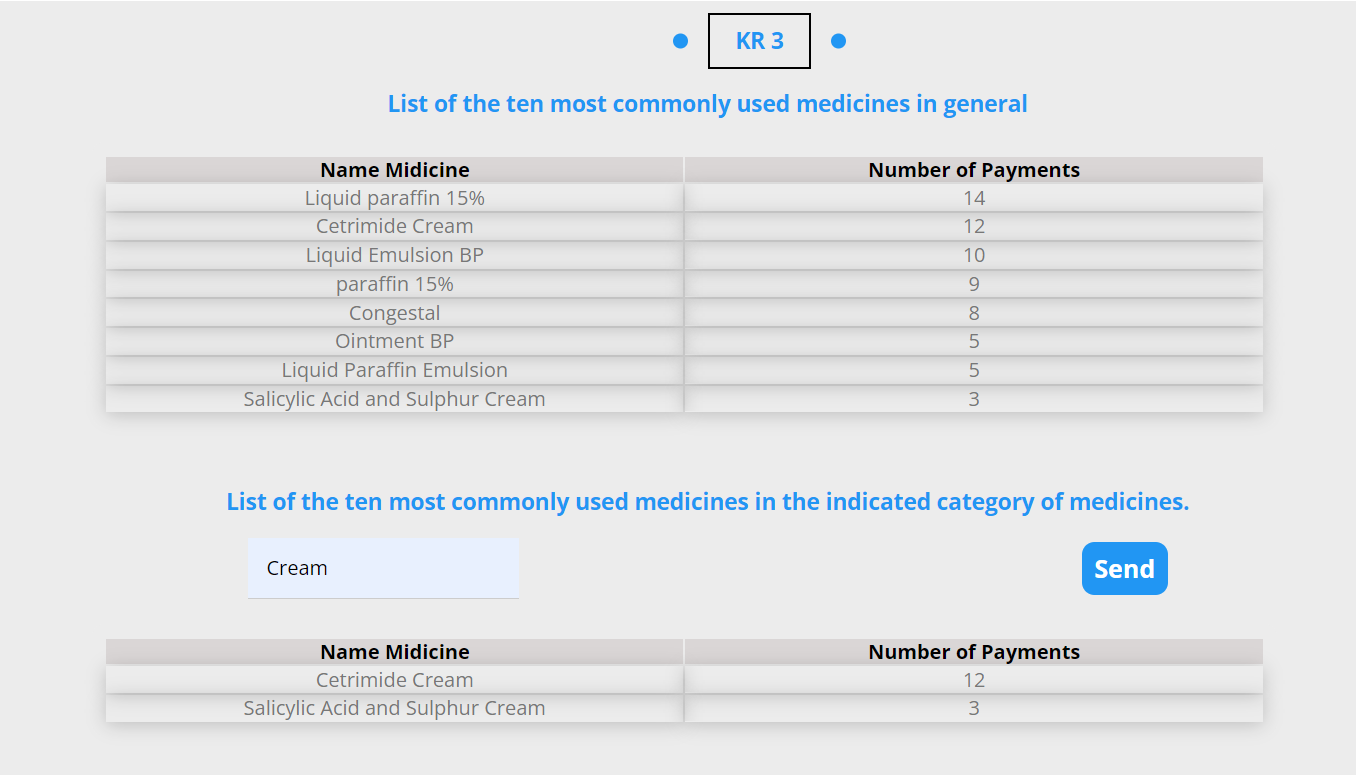
END //

DELIMITER ;



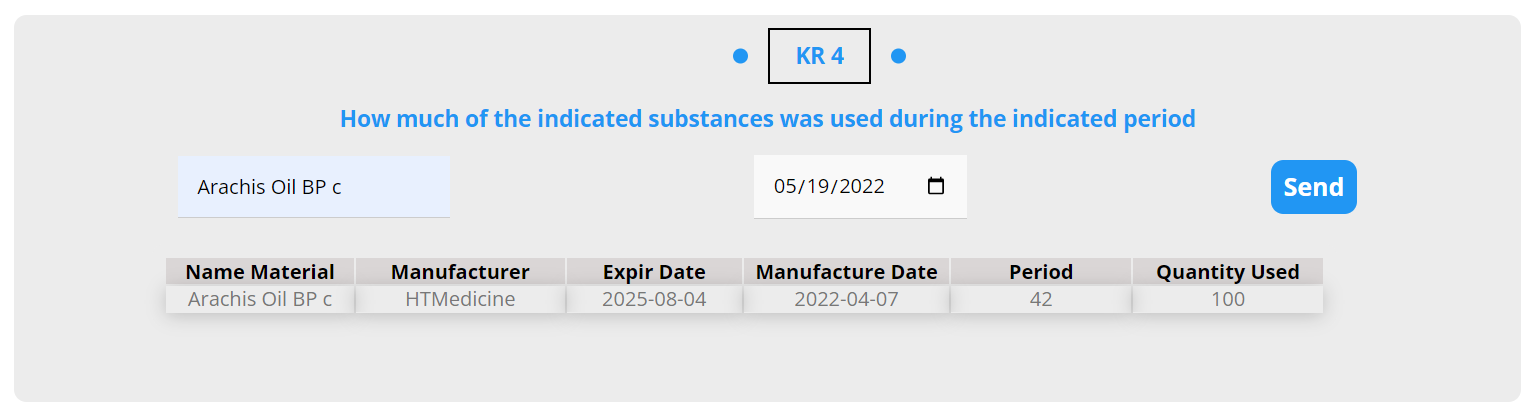
Рисунке 3.3.

1. Получите перечень десяти наиболее часто используемых медикаментов в целом и в указанной категории медикаментов,изображена на Рисунке 3.4.



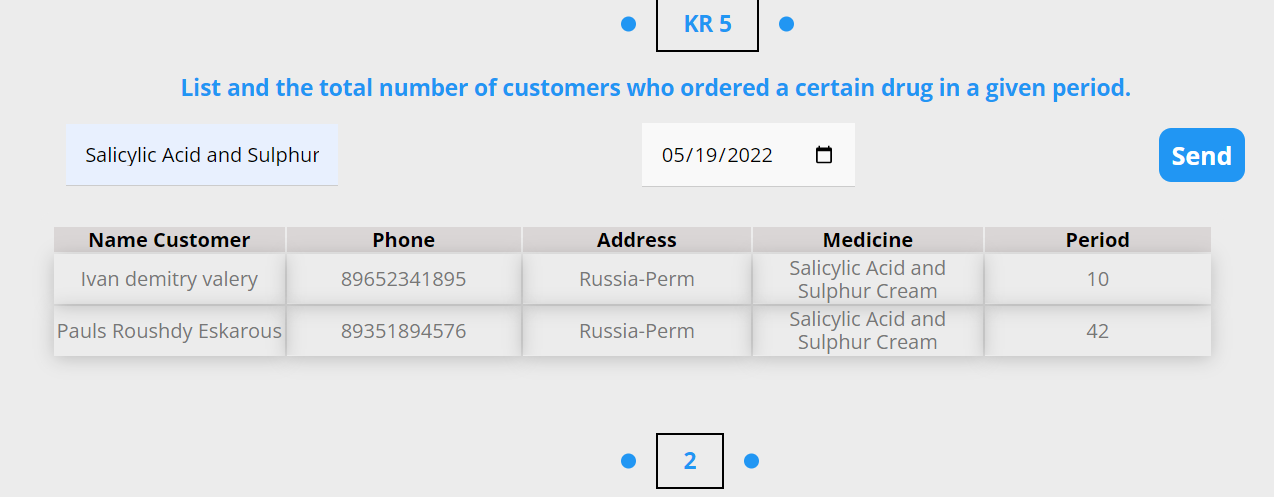
Рисунке 3.4.

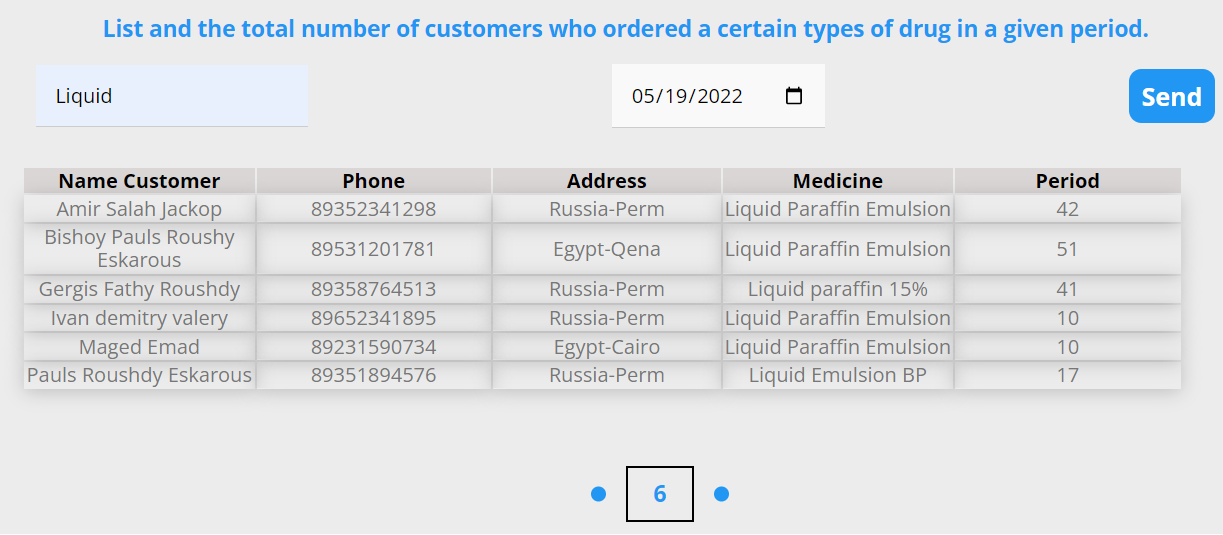
1. Подсчитайте, какой объем указанных веществ использован за указанный период,изображена на Рисунке 3.5.



Рисунке 3.5.

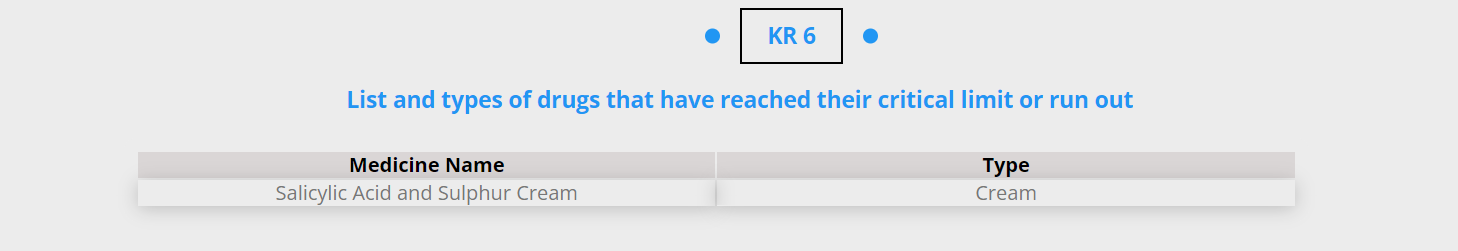
1. Получите перечень и общее число покупателей, заказывавших определенное лекарство или определенные типы лекарств за данный период,изображена на Рисунке 3.6.





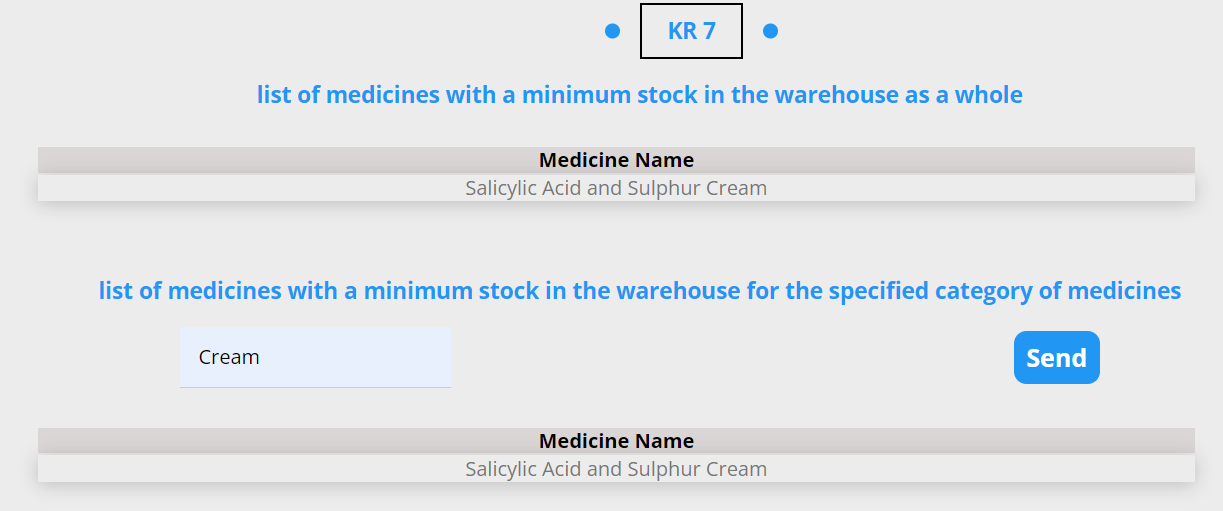
Рисунке 3.6.

1. Получите перечень и типы лекарств, достигших своей критической нормы или закончившихся,изображена на Рисунке 3.7.



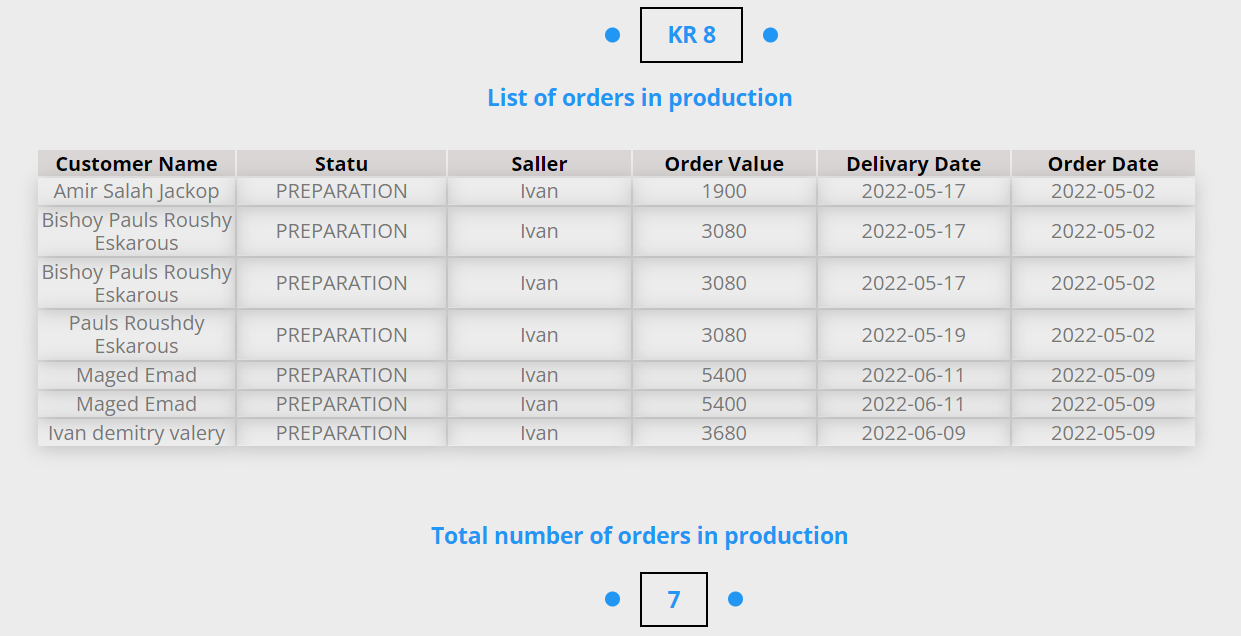
Рисунке 3.7.

1. Получите перечень лекарств с минимальным запасом на складе в целом и по указанной категории медикаментов,изображена на Рисунке 3.8.



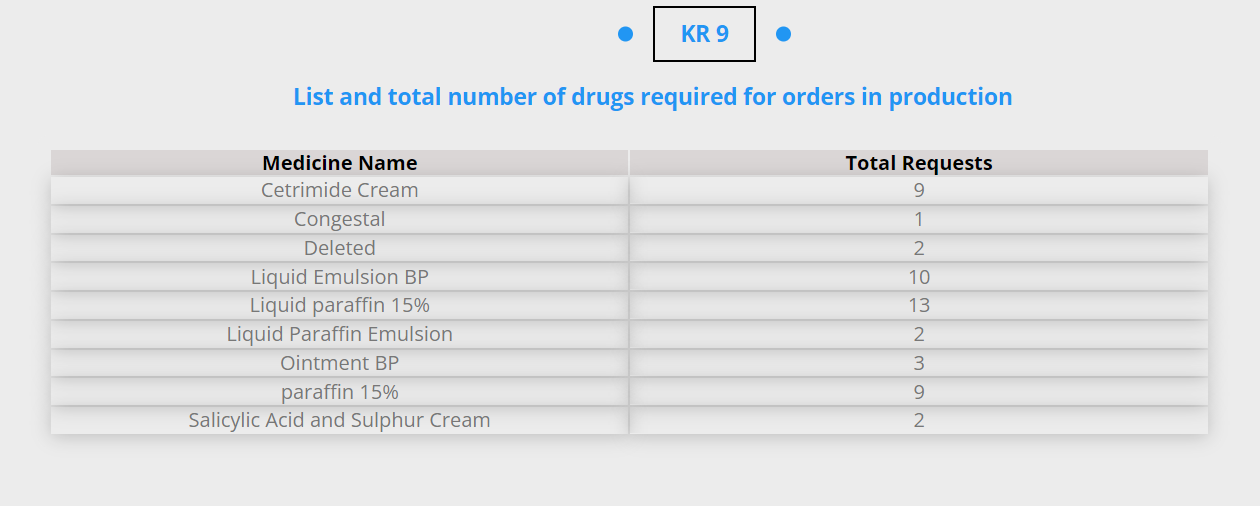
Рисунке 3.8.

1. Получите полный перечень и общее число заказов, находящихся в производстве,изображена на Рисунке 3.9.



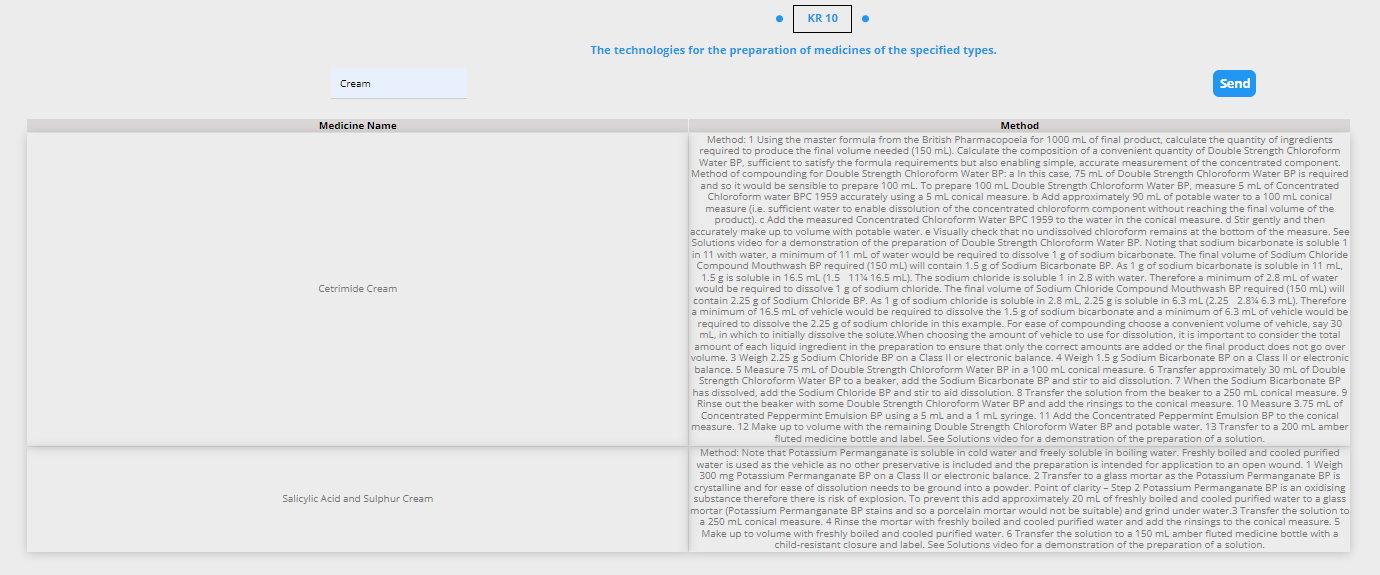
Рисунке 3.9.

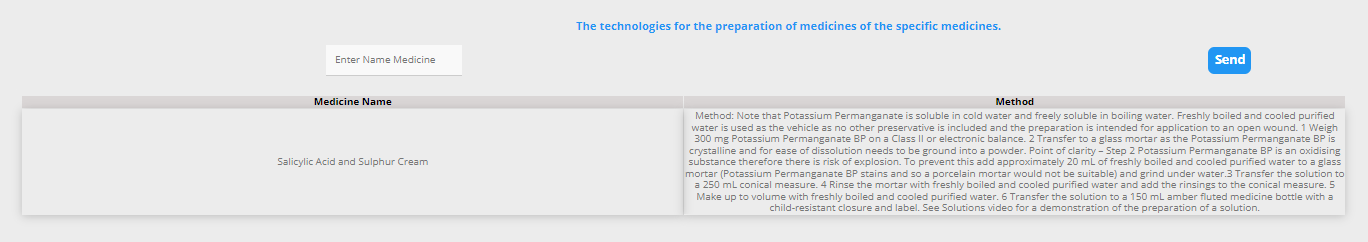
1. Получите полный перечень и общее число препаратов, требующихся заказам, находящимся в производстве,изображена на Рисунке 4.1.

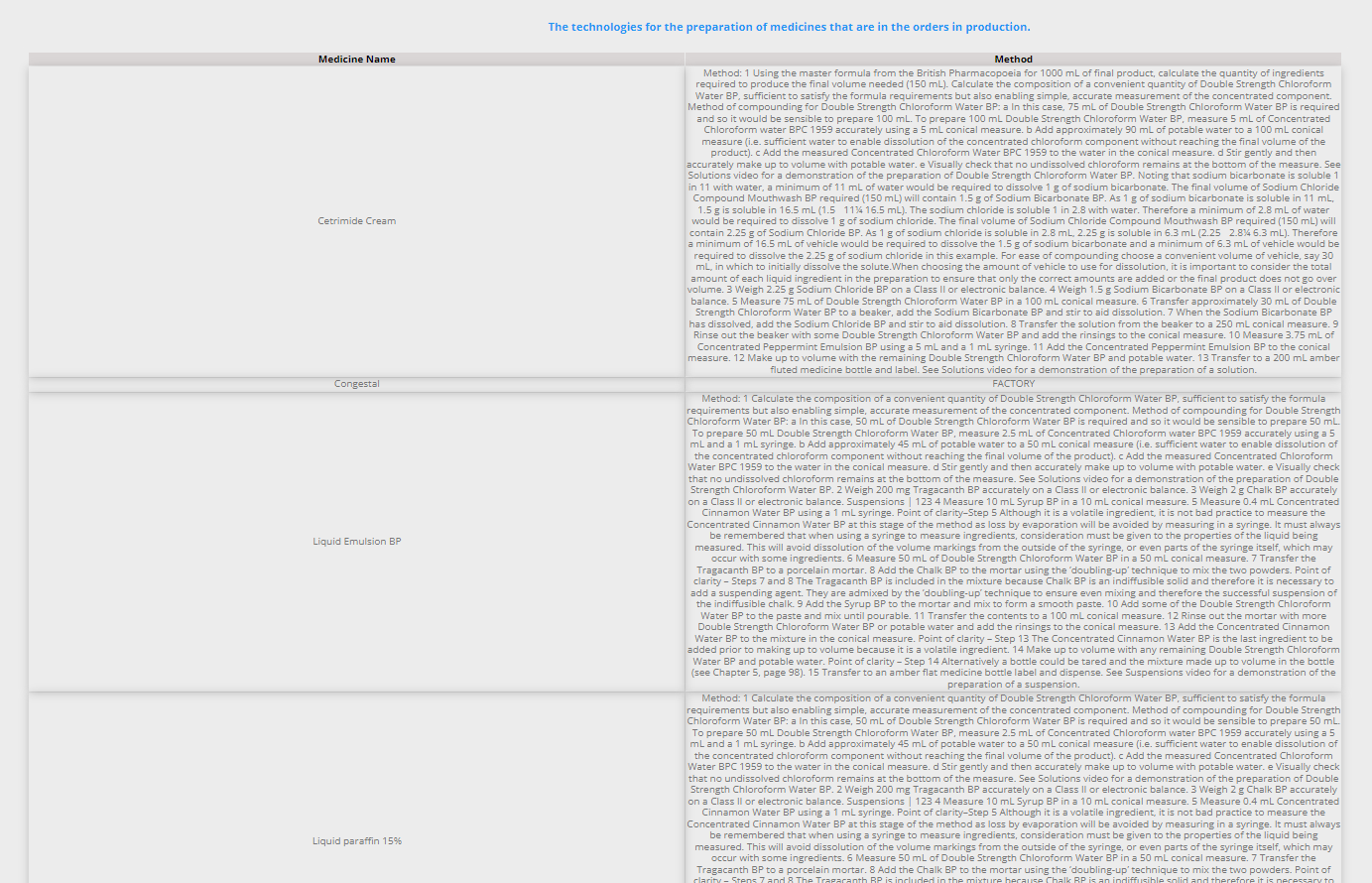


Рисунке 4.1.

1. Получите все технологии приготовления лекарств указанных типов, конкретных лекарств, лекарств, находящихся в справочнике заказов в производстве,изображена на Рисунке 4.2.

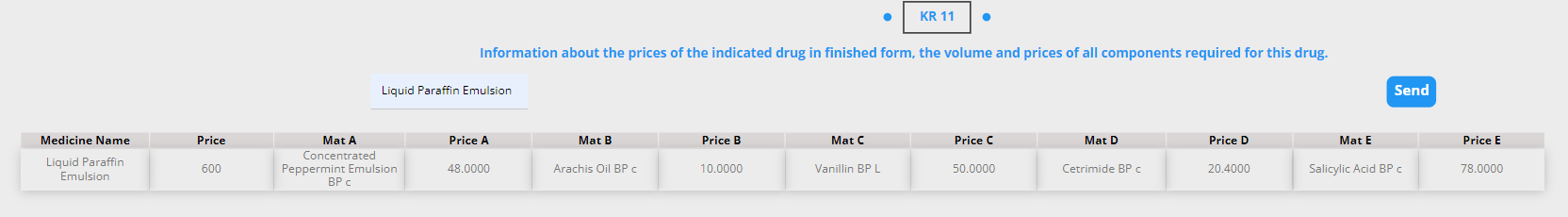






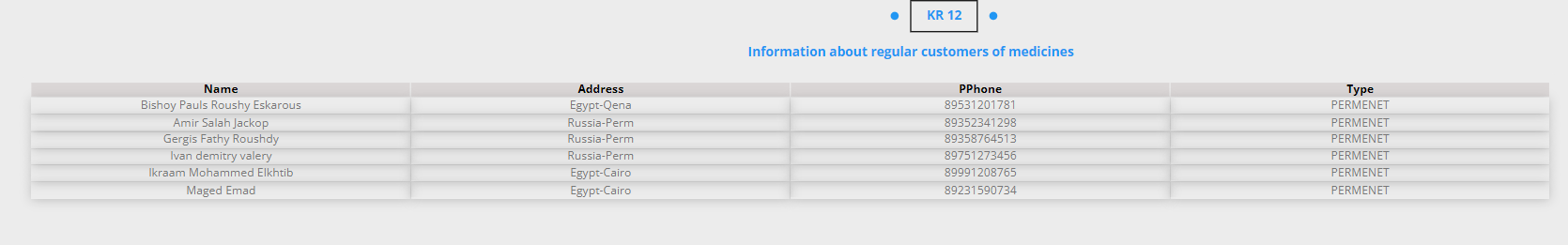
Рисунке 4.2.

1. Получите сведения о ценах на указанное лекарство в готовом виде, об объеме и ценах на все компоненты, требующиеся для этого лекарства,изображена на Рисунке 4.3.



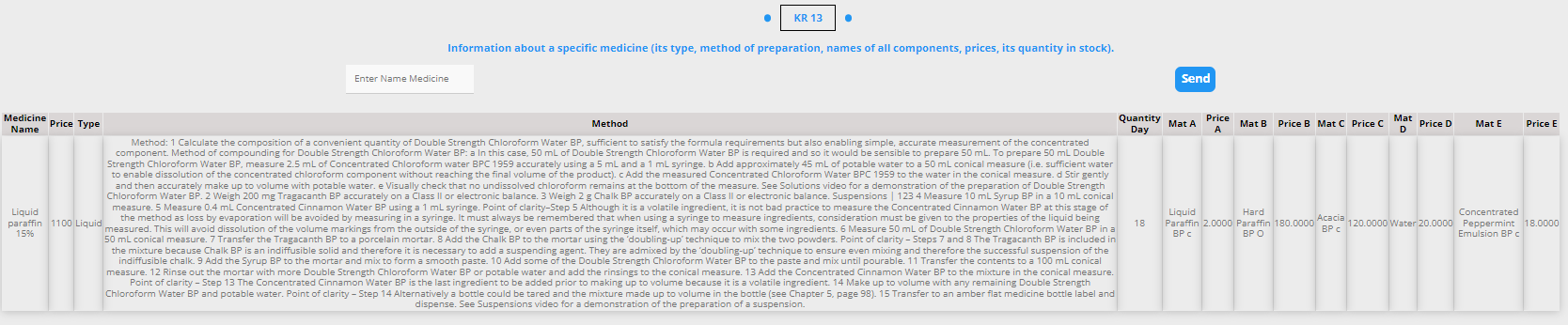
Рисунке 4.3.

1. Получите сведения о постоянных клиентах медикаментов,изображена на Рисунке 4.4.



Рисунке 4.4.

1. Получите сведения о конкретном лекарстве (его тип, способ приготовления, названия всех компонент, цены, его количество на складе),изображена на Рисунке 4.5.



Рисунке 4.5.

После тестирования основных функций программы, ошибки не были выявлены.

**Выводы**

## **3.3 Разграничение прав доступа**

Админ : имеет право доступа ко всем данным, добавления, удаления, изменения и просмотра всех данных.

**Выводы**

Программа реализована и соответствует требованиям. В ходе тестирования ошибок выявлено не было, программа корректно выполняется все запросы пользователя.

# **Заключение**

Требуемая база данных, программы успешно реализованы и отвечают требованиям задания. Модели базы данных позволили корректно реализовать связи сущностей БД и заполнить таблицы данными нужных типов. Цель работы выполнена в полном объеме.

Предоставленная база данных и веб-сайт могут быть хорошо использованы аптеками для помощи и облегчения работы.Веб-сайт позволяет аптеке корректно добавлять, удалять и изменять при работе с базой данных.

Возможность дальнейшей работы можно рассматривать как улучшение пользовательского интерфейса, использования сайта пользователем, кроме операций добавления, удаления и изменения данных непосредственно с сайта без использования системы управления базами данных СУБД.

# **Список литературы**

1. Реляционные базы данных [Электронный ресурс]

<https://www.mysqltutorial.org/>

2. Данные о лекарствах, комбинированных препаратах и сырье. Книга :Pharmaceutical Compounding and Dispensing

Автор книги:

John F Marriott

Keith A Wilson

Christopher A Langley

Dawn Belcher

1. Проектирование баз данных [Электронный ресурс]

<https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B5%D0%BA%D1%82%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%D0%B1%D0%B0%D0%B7_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85>

1. Дизайн интерфейса [Электронный ресурс]

<https://html.com/>

<https://css-tricks.com/>

1. Серверные система [Электронный ресурс]

<https://www.php.net/>

7.Инструмент для онлайн-моделирования Draw.io[Электронныйресурс]

<https://app.diagrams.net/>