МИНОБРНАУКИ РОССИИ

ФЕДЕРАЛЬНОЕ БЮДЖЕТНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«ВОСТОЧНО – СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ»

Кафедра «Программная инженерия и искусственный интеллект»

Курсовая работа

по дисциплине «Базы данных»

Тема: Разработка базы данных «Приём пациентов»

|  |  |
| --- | --- |
| Выполнил: | студент гр. Б761 |
| **\_\_\_\_\_\_\_\_\_\_** | Меркурьев З.А. |
| Руководитель: | к.т.н., доцент |
| **\_\_\_\_\_\_\_\_\_\_** | Бильгаева Л.П |
| Оценка: | **\_\_\_\_\_\_\_\_\_\_\_\_** |
| Дата защиты: | \_\_\_\_\_\_\_\_\_\_\_\_ |

Улан-Удэ

2023

ВОСТОЧНО-СИБИРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ТЕХНОЛОГИЙ И УПРАВЛЕНИЯ

Факультет компьютерных наук и технологий

Кафедра «Программная инженерия и искусственный интеллект»

ЗАДАНИЕ

на курсовую работу

|  |  |  |
| --- | --- | --- |
| **Дисциплина:** База данных | | |
| **Тема:** Разработка базы данных «Приём пациентов» | | |
| **Исполнитель:** Меркурьев З.А. | | |
| **Руководитель:** Бильгаева Л.П. | | |
| **Краткое содержание проекта:** Курсовая работа посвящена проектированию и реализации базы данных «Приём пациентов» и разработке приложения. | | |
| **1. Анализ предметной области** | | |
| 1.1 Выполнить описание предметной области | | |
| 1.2 Построить концептуальную модель предметной области | | |
| **2. Проектирование** | | |
| 2.1 Выполнить преобразование концептуальной модели в реляционную | | |
| 2.2 Выполнить нормализацию таблиц |  | |
| 2.3 Разработать запрос проектируемой базе данных | |  |
| **3. Реализация** |  | |
| 3.1 Обосновать выбор СУБД |  | |
| 3.2 Выполнить описание интерфейса | |  |
| 3.3 Выполнить описание запросов к базе данных на языке SQL | |  |
| **Сроки выполнения проекта по графику:** | | |
| 1. Теоретический раздел – 15% к 4 неделе | | |
| 1. Основной раздел. Проектирование – 100% к 6 неделе | | |
| 3. Экспериментальный раздел – 100% к 16 неделе | | |
| **Требования к оформлению:** | | |
| 1. Расчетно-пояснительная записка курсового проекта должна быть представлена в | | |
| электронной и твердой копиях. | | |
| 2. Объем РПЗ должен быть не менее 20 машинописных страниц без учета приложений. | | |
| 3. РПЗ оформляется по ГОСТу 7.32-2017. | | |

|  |  |
| --- | --- |
| Дата выдачи | 13.09.2023 |

|  |  |
| --- | --- |
| Руководитель | . |
| Исполнитель | . |

**АННОТАЦИЯ**

Данная курсовая работа посвящена вопросам проектирования и реализации базы данных «Приём пациентов». В рамках работы выполнен анализ предметной области, разработана концептуальная модель предметной области и выполнено ее преобразование в реляционную модель. Разработаны логическая и физическая модели данных. Разработано и реализовано программное приложение на языке программирования C#, приведено его описание.

Данная курсовая работа состоит из введения, трех основных разделов (анализа предметной области, проектирования, реализации), заключения и списка использованных источников.

Курсовая работа содержит страницы, таблицы, рисунков и источников.

**СОДЕРЖАНИЕ**

[1 Анализ предметной области 7](#__RefHeading___Toc1101_1501111604)

[1.1 Описание предметной области 7](#__RefHeading___Toc1103_1501111604)

[1.2 Разработка концептуальной модели предметной области 7](#__RefHeading___Toc1105_1501111604)

[2 Проектирование 10](#__RefHeading___Toc1107_1501111604)

[2.1 Преобразование концептуальной модели в реляционную 10](#__RefHeading___Toc1109_1501111604)

[2.2 Нормализация таблиц 10](#__RefHeading___Toc1111_1501111604)

[2.3 Разработка запросов к БД 10](#__RefHeading___Toc1113_1501111604)

[3 Реализация 11](#__RefHeading___Toc1115_1501111604)

[3.1 Обоснование и выбор СУБД 11](#__RefHeading___Toc1117_1501111604)

[3.2 Описание физической модели базы данных 11](#__RefHeading___Toc1119_1501111604)

[СОДЕРЖАНИЕ 4](#__RefHeading___Toc1097_1501111604)

[3.4 Реализация запросов 11](#__RefHeading___Toc1123_1501111604)

[ЗАКЛЮЧЕНИЕ 12](#__RefHeading___Toc1125_1501111604)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 13](#__RefHeading___Toc1127_1501111604)

# ВВЕДЕНИЕ

Современное состояние информационных технологий можно охарактеризовать наличием большого количества функционирующих баз данных, содержащих информацию практически по всем видам деятельности общества. Отличительной чертой базы данных является то, что информация в ней систематизирована по определенным правилам и может быть обработана с использованием различных алгоритмов. Информация может храниться в базе данных до тех пор, пока она актуальная.

Актуальность темы данной курсовой работы заключается в том, что современные клинки ведут у не могут осуществлять свою деятельность без использования баз данных, так как они являются основным инструментом для учета клиентов. Применение базы данных и программного приложения позволит автоматизировать задачи хранения и обработки данных, используемых в частной клинике.

Целью курсовой работы является разработка базы данных «Приём пациентов».

Для достижения поставленной цели необходимо решить следующие задачи:

* выполнить анализ предметной области;
* разработать концептуальную модель предметной области;
* разработать логическую модель данных;
* разработать транзакции к проектируемой базе данных;
* выполнить реализацию базы данных в среде СУБД MS SQL Server;
* разработать программное приложение на языке программирования C#.

Расчетно-пояснительная записка состоит из введения, трех основных разделов (анализа предметной области, проектирования и реализации), заключения и списка использованных источников.

В первом разделе приведено описание предметной области и разработки концептуальной модели предметной области.

Во втором разделе разработана логическая модель данных путем преобразование концептуальной модели в реляционную и нормализации таблиц, разработаны запросы к базе данных.

В третьем разделе выполнено обоснование и выбор СУБД, реализована база данных, описан интерфейс программного приложения и реализация запросов к базе данных на языке SQL.

# 1 Анализ предметной области

## **1.1 Описание предметной области**

Разрабатываемая модель базы данных предоставляет работу части функционала поликлиники, а именно приём пациентов.

Поликлиника – это многопрофильное лечебно-профилактическое учреждение, оказывающее медицинскую помощь населению на закрепленной территории на до госпитальном этапе.

Число отделений и кабинетов, их потенциальные возможности определяются мощностью поликлиники и количеством штатных должностей, которые зависят от численности закрепленного за поликлиникой населения. Структура поликлиники (открытие тех или иных отделений, кабинетов и т. п.) зависит от обращаемости населения в это учреждение, от способности поликлиники предоставить больным необходимую медицинскую помощь.

Приём самих клиентов организован следующим образом: Клиент, пришедший в поликлинику, обращается в регистратуру, и регистрируется. Дальше обращается к определённому доктору и проходит на приём.

База данных должна в полной мере осуществлять все задуманные процессы: регистрация приходящих пациентов, запись даты приёма, времени приёма, номера приёма, результата приёма. А также должна быть полезна и давать возможность предоставлять полную информационную систему для групп пользователей: главного врача поликлиники и потенциального проверяющего лица. Необходимо, чтобы база данных по мере внесения новых записей в систему вовремя обновлялась, включая в себя следующую информацию:

1. Информация, которая будет характеризовать новых врачей, их специализация и дата поступления и номер закреплённого за ними кабинета.
2. Информация, характеризующая поступающих пациентов – фамилия, имя, отчество, дата рождения, адрес пациентов, телефон, номер мед. книжки;
3. Информация о результатах приёма клиентов.

## **1.2 Разработка концептуальной модели предметной области**

Разрабатываемая модель базы данных предоставляет работу части функционала поликлиники, а именно приём пациентов.

Концептуальная модель базы данных это некая наглядная диаграмма, нарисованная в принятых обозначениях и подробно показывающая связь между объектами и их характеристиками. Создается концептуальная модель для дальнейшего проектирования базы данных и перевод ее, например, в реляционную базу данных. На концептуальной модели в визуально удобном виде прописываются связи между объектами данных и их характеристиками.

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

1. Объект или сущность

Это фактически вещь или объект за которым пользователь будет наблюдать.

2. Атрибут

Это характеристика объекта, соответствующая его сущности.

3. Связь или отношения между объектами.

Средством моделирование предметной области на этапе концептуального проектирование является модель “Сущность-связь» (ER-модель). В ней моделирование структуры данных предметной области базируются на использовании графических средств ER-диаграмм.

Основными понятиями ER-диаграммы являются сущность, атрибут, связь. Сущность – это некоторый объект реального мира, который может существовать независимо. Атрибут – это свойство сущности. Связь – это представление взаимодействия между сущностями

На основе изучения и анализа предметной области выявлены следующие сущности: ВРАЧ, УСЛУГА, ПАЦИЕНТ, ПРИЁМ.

Сущность ВРАЧ — содержит информацию о враче, фамилия, имя, Отчество, дата поступления на работу, номер кабинета, специализация, табельный номер.

Сущность УСЛУГА — содержит информацию о услуге, код услуги и стоимость.

Сущность ПАЦИЕНТ — содержит номер мед. Карты, фамилия, имя, отчество, дата рождения, адрес жительства, адрес, телефон.

Сущность ПРИЁМ — содержит дату приёма, время приёма, номер мед. Карты пациента, табельный номер врача, код услуги, результат приёма.

Далее необходимо установить взаимосвязи между отдельными информационными объектами, что осуществляется путем анализа типов связей между ключами с учетом сформулированных ранее условий описания предметной области.

В рассматриваемой предметной области можно выделить 2 типа связи.

1. Врач — осуществляет — Приём

Один врач может осуществлять множество приёмов. Следовательно связь имеет тип «один-ко-многим». Каждый экземпляр сущности Врач связан с экземпляром сущности Приём также, как и каждый экземпляр сущности Приём связан с экземпляром сущности Врач, поэтому классы принадлежностей обеих сущностей являются обязательными. ER-диаграмма связи Врач-Приём представлена на рисунке 1.1.

Приём

Врач

Осуществляет

Рисунок 1.1 — Связь между Врач - Приём

2. Пациент - проходит — Приём

Поскольку пациент может проходить неограниченное количество приёмов. Следовательно связь имеет тип «один-ко-многим».Каждый экземпляр сущности Пациент связан с экземпляром сущности Приём также, как и каждый экземпляр сущности Приём связан с экземпляром сущности Пациент, поэтому классы принадлежностей обеих сущностей являются обязательными. ER-диаграмма связи Пациент-Приём представлена на рисунке 1.2.

1

м

Приём

Пациент

проходит

Рисунок 1.2 — Связь между Врач — Приём

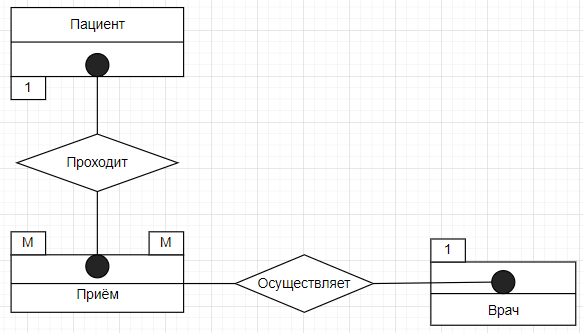


Рисунок 1.3 — ER модель

Атрибутный состав сущностей представлен в таблицах 1.1 — 1.4

Таблица 1.1 – Атрибутный состав сущности Доктор

|  |
| --- |
| **Доктор** |
| **Код доктора** |
| ФИО доктора |
| Дата поступления на работу |
| Номер кабинета |
| Специализация |

Таблица 1.2 – Атрибутный состав сущности Пациент

|  |
| --- |
| **Пациент** |
| Код пациента |
| ФИО |
| Дата рождения |
| Адрес |
| Номер телефеона |

Таблица 1.3 – Атрибутный состав сущности Приём

|  |
| --- |
| **Приём** |
| Код Доктора |
| Код Пациента |
| Код Результата Приёма |

Таблица 1.4 – Атрибутный состав сущности Услуга

|  |
| --- |
| **Услуга** |
| **Код услуги** |
| Цена услуги |

Таблица 1.5 – Атрибутный состав сущности Результат Приёма

|  |
| --- |
| **Услуга** |
| **Код результата приёма** |
| Описание |
| Рекомендации |
| Диагноз |
| Выписанные лекарства |
| Оказанные услуги |
| Информация о докторе |
| Информация о пациенте |

Таблица 1.6 – Атрибутный состав сущности Услуги врача

|  |
| --- |
| **Услуги врача** |
| Код доктора |
| Код услуги |

# 2 Проектирование

## **2.1 Преобразование концептуальной модели в реляционную**

Концептуальные модели позволяют более точно представить предметную область, чем реляционные и другие более ранние модели. Но в настоящее время существует немного систем управления базами данных, поддерживающих эти модели. На практике наиболее распространены системы, реализующие реляционную модель. Поэтому необходим метод перевода концептуальной модели в реляционную. Такой метод основывается на формировании набора предварительных таблиц из ER-диаграмм.

Для каждой сущности создаётся таблица. Причём каждому атрибуту сущности соответствует столбец таблицы.

Для создания таблиц необходимо определить правила генерации таблиц из ER-диаграмм. Правила опираются на два основных фактора – тип связи и класс принадлежности сущности.

Таким образом, для сущностей Пациент и Приём, имеющих связь один-ко-многим, по правилу преобразования 4 необходимо сгенерировать таблицы для каждый сущности, представленные на рисунках 2.1 и 2.2 соответственно.



Рисунок 2.1 Таблица Пациент



Рисунок 2.2 Таблица Приём

Эти таблицы будут связаны так, как показано на рисунке 2.3

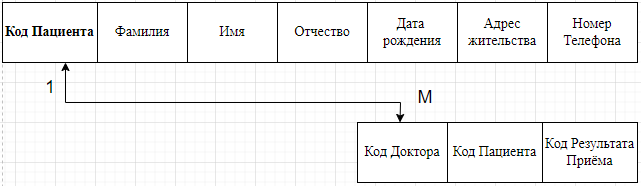


Рисунок 2.3 Таблица Пациент-Приём

Для сущностей Доктор и Приём, имеющих связь один-ко-многим, по правилу преобразования 4 необходимо сгенерировать таблицы для каждый сущности, представленные на рисунках 2.4 и 2.5 соответственно.



Рисунок 2.4 Таблица Доктор



Рисунок 2.5 Таблица Приём

Эти таблицы будут связаны так, как показано на рисунке 2.6

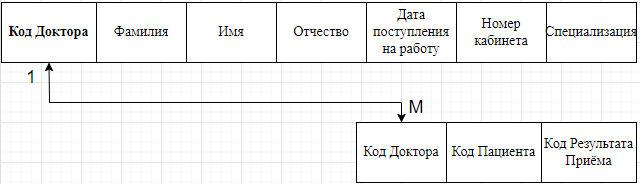


Рисунок 2.6 Таблица Доктор - Приём

Для сущностей Услуги доктора и Доктор, имеющих связь один-ко-многим, необходимо сгенерировать таблицы для каждый сущности, представленные на рисунках 2.7 и 2.8 соответственно.

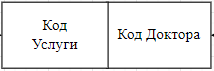


Рисунок 2.7 Таблица Услуги доктора

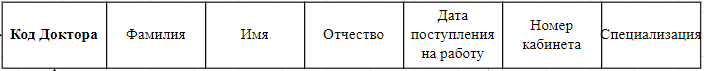


Рисунок 2.8 Таблица доктор

Эти таблицы будут связаны так, как показано на рисунке 2.9



Рисунок 2.9 Таблица Доктор — Услуги доктора

Для сущностей Услуги доктора и Доктор, имеющих связь один-ко-многим, необходимо сгенерировать таблицы для каждый сущности, представленные на рисунках 2.10 и 2.11 соответственно.

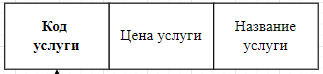


Рисунок 2.10 - Услуги

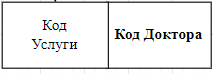


Рисунок 2.11 Таблица Услуги доктора

Эти таблицы будут связаны так, как показано на рисунке 2.12

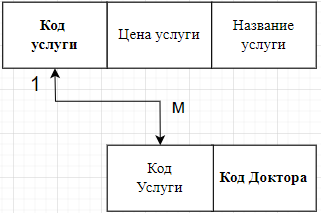


Рисунок 2.12 - Таблица Доктор — Услуги доктора

Для сущностей Приём и Результат приёма, имеющих связь один-к-одному, необходимо сгенерировать таблицы для каждый сущности, представленные на рисунках 2.13 и 2.14 соответственно.

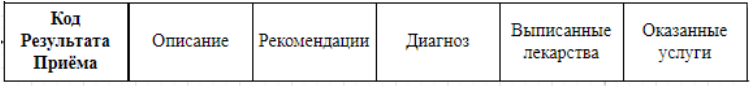


Рисунок 2.13 - Таблица Результат Приёма

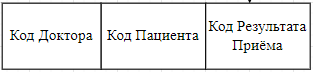


Рисунок 2.14 - Таблица Приём

Эти таблицы будут связаны так, как показано на рисунке 2.15

Рисунок 2.15 — Таблица Результат Приёма - Приём

Таблицы связываются посредством связи первичных и внешних ключей. В результате получим реляционную модель для ER-модели предметной области«Приём пациентов», представленную на рисунке 2.16.

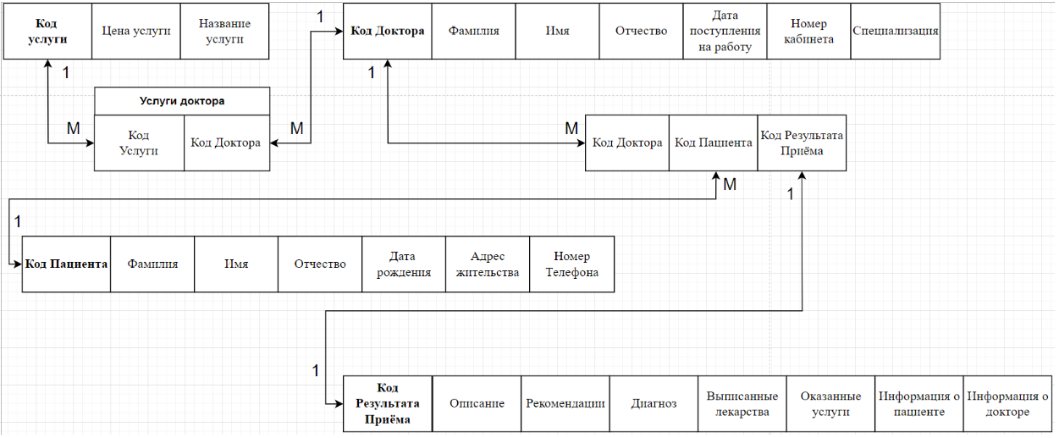


Рисунок 2.16 — Реляционная модель

## **2.2 Нормализация таблиц**

Нормализация таблиц — это процесс организации данных в базе данных, включающий создание таблиц и установление отношений между ними в соответствии с правилами, которые обеспечивают защиту данных и делают базу данных более гибкой, устраняя избыточность и несогласованные зависимости. Реляционная база данных считается эффективной, если все ее таблицы находятся как минимум в 3 нормальной форме (3НФ).

Таблица находится в 1НФ, если все ее поля содержат только простые неделимые значения. Следует найти и устранить все повторяющиеся группы данных.

Таблица находится в 2НФ, если она удовлетворяет требованиям 1НФ и каждый атрибут, не входящий в состав первичного ключа, характеризуется функционально полной зависимостью от этого первичного ключа. Функционально полная зависимость означает, что атрибут функционально зависит от всего первичного составного ключа, но при этом не находится в функциональной зависимости от какой-либо из входящих в него атрибутов (частей). Или другими словами: в 2НФ нет неключевых атрибутов, зависящих от части составного ключа.

Таблица находится в 3НФ, если она удовлетворяет требованиям 2НФ и не содержит транзитивных зависимостей неключевых атрибутов от ключевых. Любой не ключевой атрибут зависит только от первичного ключа.

Результат произведенной нормализации таблиц реляционной модели данных представлен на рисунке 2.17.

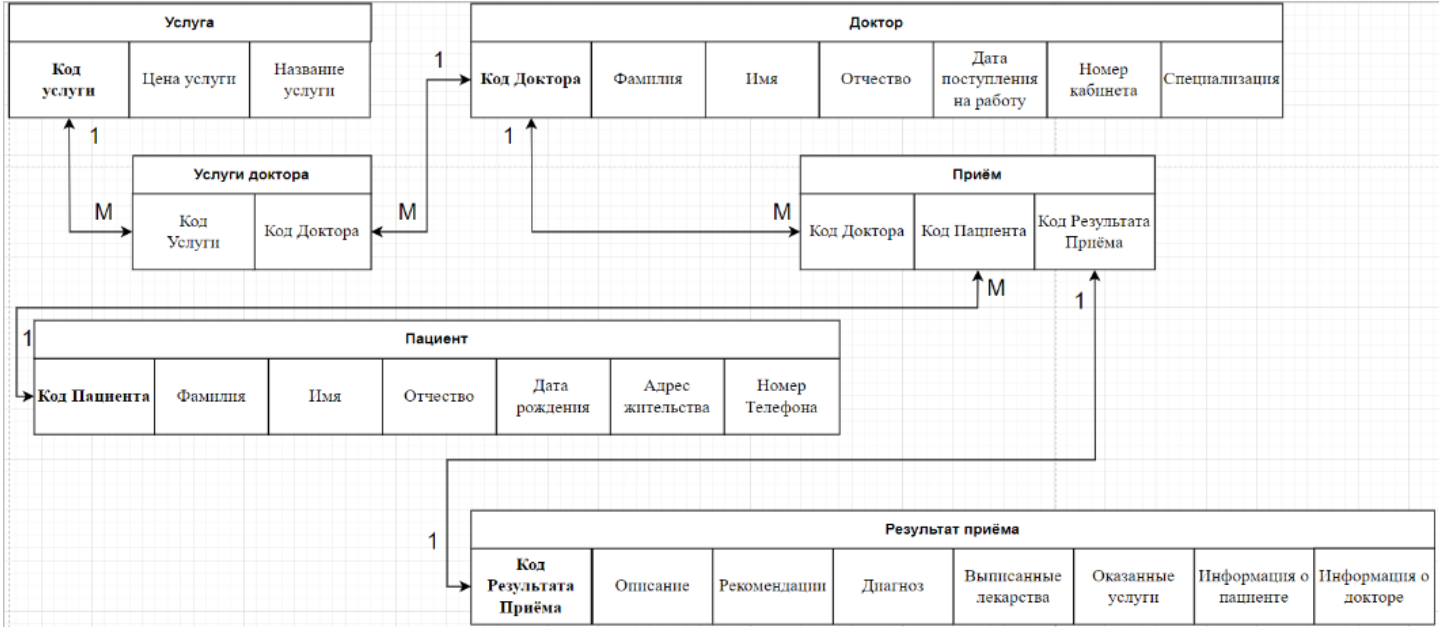


Рисунок 2.17 - Логическая модель данных

Логическая модель состоит из шести сущностей. Атрибутный состав каждой из них приведён ниже.

Сущность Доктор содержит атрибуты (рис.2.18)

- Код доктора — уникальный идентификатор доктора

- Фамилия, Имя, Отчество Доктора

- Дата поступления на работу

- Номер кабинета

- Специализация



Рисунок 2.18 — Сущность доктор

Сущность Пациент содержит атрибуты (рис.2.19)

- Код пациента— уникальный идентификатор пациента

- Фамилия, Имя, Отчество пациента

- Дата рождения

- Адрес жительства

- Номер телефона

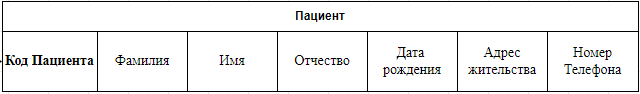


Рисунок 2.19 — Сущность пациент

Сущность Приём содержит атрибуты (рис.2.20)

- Код доктора — идентификатор доктора, который ведёт приём

- Код пациента — идентификатор пациента, который проходит приём

- Код результата приёма — идентификатор объекта, который будет характеризовать результат приёма.

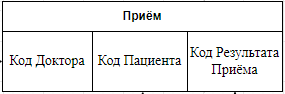


Рисунок 2.20 — Сущность приём

Сущность Услуга содержит атрибуты (рис.2.21)

- Код услуги — уникальный идентификатор услуги

- Цена услуги

- Название услуги

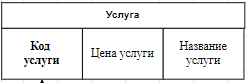


Рисунок 2.21 — Сущность Услуга

Сущность Услуги Доктора содержит атрибуты (рис.2.22)

- Код услуги — идентификатор объекта, который характеризует услугу.

- Код доктора — идентификатор объекта, который характеризует доктора, предоставляющего данную услугу.

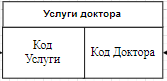


Рисунок 2.22 — Сущность Услуги Доктора

Сущность Результат Приёма содержит атрибуты (рис.2.23)

- Код результата услуги — уникальный идентификатор результата услуги.

- Описание

- Рекомендации доктора

- Диагноз

- Выписанные лекарства

- Оказанные услуги

- Информация о пациенте

- Информация о докторе

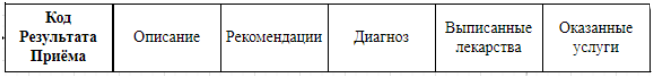


Рисунок 2.23 — Сущность Результат Приёма

## **2.3 Разработка запросов к БД**

Запрос – операция, выполняемая на языке SQL и позволяющая извлекать данные из базы по различным критериям. Запрос позволяет выбрать необходимые данные из одной или не-скольких взаимосвязанных таблиц, произвести вычисления и получить результат в виде виртуальной таблицы. Полученная таблица может использоваться в качестве источника данных в следующих запросах.

В проектируемой базе данных необходимо реализовать следующие запросы:

- Вывод списка докторов, чья специализация ортопедия.

- Вывод списка пациентов, кто старше 70.

- Вывод списка результатов приёма, где было оказано больше двух услуг.

- Вывод списка всех пациентов.

- Вывод списка услуг стоимостью меньше 5000 р

# 3 Реализация

## **3.1 Обоснование и выбор СУБД**

Для хранения данных разработана база данных, реализованная в СУБД SQL Server 2019.

СУБД SQL Server 2019 является эффективным инструментом для работы с большим объемом данных, дает возможность формировать запросы и осуществлять поиск, синхронизировать информацию, выполнять аналитическую обработку данных и получать разнообразные отчеты.

Все базы данных в этом СУБД хранятся в файлах типа .mdf вместе с журналом транзакций, имеющих тип .ldf. Количество баз данных, а также таблиц в них, ограничено только свободным местом, имеющимся на жестком диске.

СУБД MS SQL Server 2019 имеет следующие преимущества:

* масштабирование системы. Взаимодействовать с ней можно как на простых ноутбуках, так и на ПК с мощным процессором, который способен обрабатывать большой объем запросов.
* данные извлекаются быстро, а сложную информацию удобнее хранить. Система обрабатывает транзакции в интерактивном режиме, есть динамическая блокировка.
* автоматизация рутинных административных задач. Например, управление блокировками и памятью, редактура размеров файлов. В программе продуманы настройки, можно создавать профили пользователей;
* удобный поиск. Его можно осуществлять по фразам, словам, тексту либо создавать ключевые индексы;
* поддержка работы с другими решениями Майкрософт, в том числе с Excel, Access.

## **3.2 Описание физической модели базы данных**

Физическое проектирование базы данных – это процесс подготовки описания реализации базы данных на вторичных запоминающих устройствах; на данном этапе рассматриваются основные отношения, организация файлов и индексов, предназначенных для обеспечения эффективного доступа к данным. Цель физического проектирования – описание способа физической реализации логической модели базы данных.

Физическая модель данных рассматриваемой предметной области представлена на рисунке 3.1.

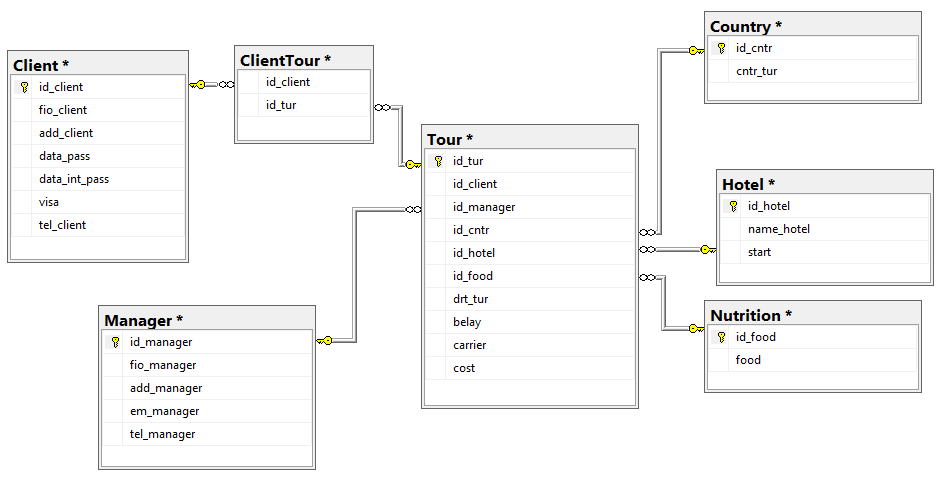


Рисунок 3.1 – Физическая модель данных

Структуры таблиц представлены на рисунках 3.2-3.6.

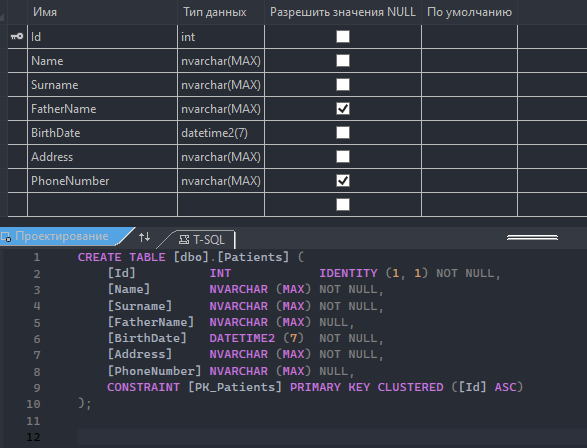


Рисунок 3.2 – Структура таблицы «Пациент»

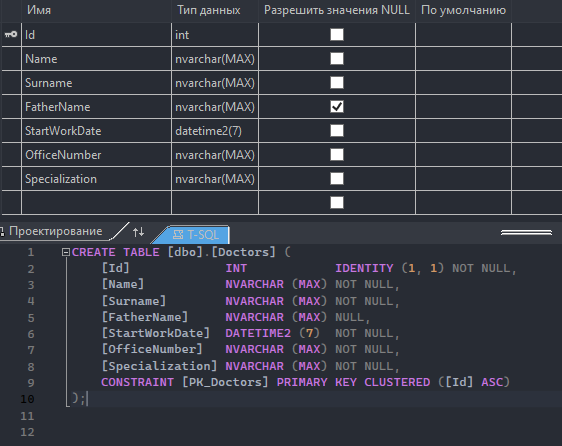


Рисунок 3.3 – Структура таблицы «Доктор»

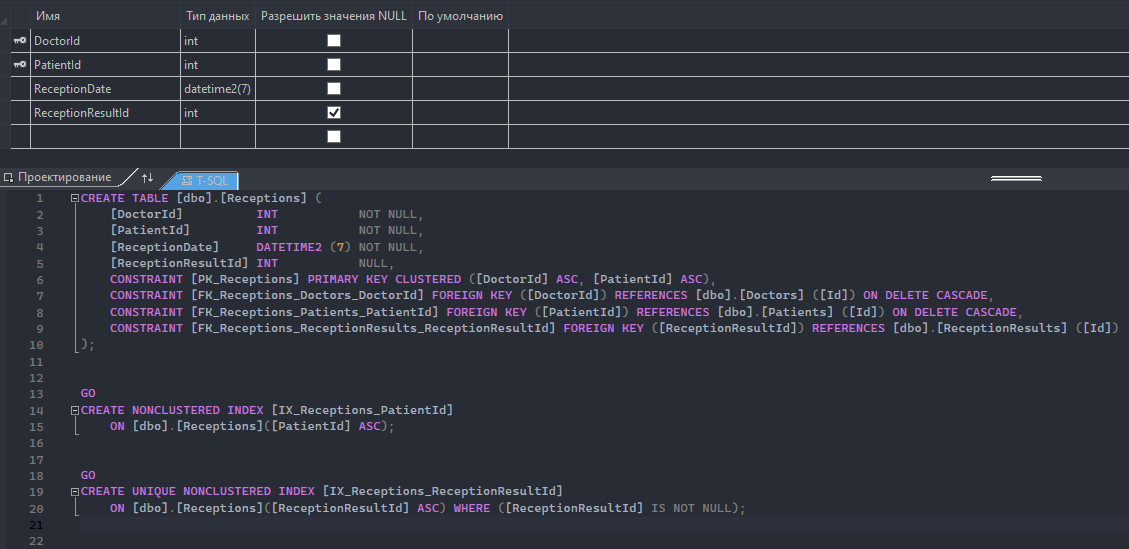


Рисунок 3.4 – Структура таблицы «Приём»

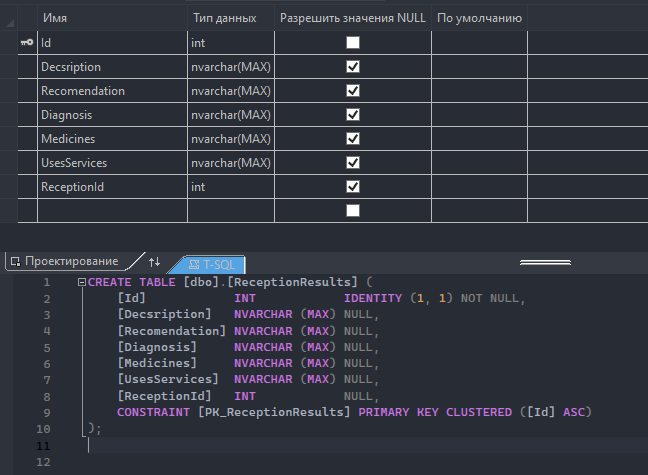
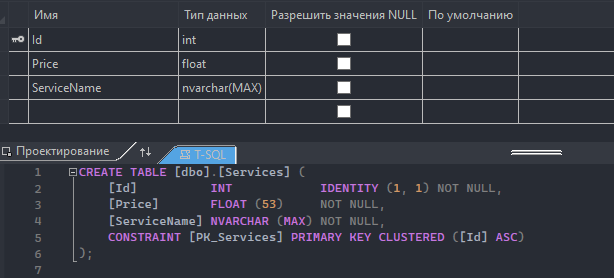


Рисунок 3.5 – Структура таблицы «Результат приёма»

Рисунок 3.6 – Структура таблицы «Услуга»

## **3.3 Описание интерфейса**

## **3.4 Реализация запросов**

# ЗАКЛЮЧЕНИЕ

# СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ