

Муром 2024 г

В данной курсовой работе была разработана база данных для поликлиники, а также приложение для взаимодействия с ней. В процессе выполнения работы был проведён анализ технического задания, спроектированы структура базы данных и архитектура программы, а также основные алгоритмы.

Для разработки использовался язык программирования C# и среда разработки Visual Studio 2022. В качестве системы управления базами данных (СУБД) была выбрана Firebird версии 2.5.9.

Содержание

Введение	4
1. Анализ технического задания	6
2. Разработка моделей данных	7
3. Разработка и реализация ИС	12
4. Тестирование ИС	17
Заключение	20
Список используемой литературы	21
Приложение 1. Модели данных	22
Приложение 2. Текст программы	24
Приложение 3. Снимки окон программы	25

					МИВУ 01.03.02							
Изм.	Лист	№ докум.	Подпись	Дат	Курсовая работа АИС поликлиники			Лит.	Лист	Листов		
Провер.		Колпаков А.А.									3	28
Разраб.		Косолапова Е.А.						МИ ВлГУ ПМИ-112				
Реценз.												
Н. Контр.												
Утверд.		Орлов А.А.										

Введение.

Современные медицинские учреждения, такие как поликлиники, сталкиваются с необходимостью оптимизации своих процессов и повышения качества обслуживания пациентов. В условиях стремительного развития информационных технологий и увеличения объем данных, которые необходимо обрабатывать, становится очевидной важность внедрения автоматизированных информационных систем (АИС). Эти системы позволяют не только упростить и ускорить работу медицинского персонала, но и обеспечить более высокий уровень взаимодействия с пациентами, что в свою очередь способствует улучшению качества медицинских услуг. В данной курсовой работе будет рассмотрено проектирование информационной системы для поликлиники, что является актуальной задачей в свете современных требований к здравоохранению.

Целью данной курсовой работы является разработка автоматизированной информационной системы для поликлиники.

Основные задачи:

1. Разработка концептуальной, логической, физической моделей базы данных.
2. Реализация физической модели в базе данных.
3. Написание обработчиков событий и функций для работы с данными в БД.

Актуальность данной работы обусловлена тем, что в последние годы наблюдается значительный рост интереса к автоматизации процессов в здравоохранении. Поликлиники, как важные звенья в системе здравоохранения, нуждаются в эффективных инструментах для управления данными о пациентах, записи на приём, учёта медицинских услуг и многого другого. Внедрение АИС позволяет не только сократить время на выполнение рутинных операций, но и минимизировать вероятность ошибок, связанных с человеческим фактором. Кроме того, автоматизация процессов

					МИВУ 01.03.02	Лист
Изм.	Лист	№ докум.	Подпись	Дат		4

способствует более эффективному использованию ресурсов и повышению общей производительности работы учреждения.

Важным аспектом является разработка моделей, которые помогут визуализировать структуру и функциональность системы.

SQL является стандартным языком для работы с реляционными базами данных и позволит эффективно управлять данными, обеспечивая их целостность и доступность. В рамках работы будет рассмотрено, как с помощью SQL можно реализовать необходимые функции для работы с данными о пациентах, врачах, записях на приём и других аспектах.

Тестирование и отладка информационной системы также займут важное место в данной работе. Эти этапы необходимы для выявления и устранения ошибок, а также для проверки соответствия системы заявленным требованиям.

Таким образом, данная курсовая работа охватывает спектр вопросов, связанных с проектированием автоматизированной информационной системы для поликлиники, и направлена на решение актуальных задач, стоящих перед современным здравоохранением.

					МИВУ 01.03.02	Лист
						5
Изм.	Лист	№ докум.	Подпись	Дат		

1. Анализ технического задания.

В данной курсовой работе поставлена задача создания автоматизированной информационной системы поликлиники. Для разработки такой системы необходимы СУБД для создания базы данных и среда разработки для создания интерфейса и функций по работе с БД.

Согласно заданию, в программе нужно учесть следующие особенности:

- Для каждого доктора должен быть доступен список пациентов, которые его посетили за заданный период времени

- Иметь возможность автоматически генерировать отчёты для анализа результатов работы поликлиники

- В результате визита врач ставит пациенту диагноз

В поликлинику приходят пациенты для получения медицинских услуг, консультаций и диагностики. Во время визита пациента врач проводит осмотр и ставит диагноз.

Исходя из данных требований к приложению, база данных должна содержать несколько таблиц и данных таблицы должны быть связаны для целостности системы.

Для того чтобы создать программу, необходимо учесть то, что она создаётся, прежде всего, для пользователя, и поэтому немаловажным требованием к программе должен стать удобный и интуитивно понятный интерфейс. Необходимо предусмотреть все возможности управления приложением, чтобы упростить работу пользователя и максимально обеспечить эффективность работы.

Программа должна правильно работать с данными, т.е всегда должен выводиться нужный результат, требуемый пользователю. Приложение должно мгновенно реагировать на действия пользователя и в зависимости от запроса с его стороны формировать выходной результат.

					МИВУ 01.03.02	Лист
						6
Изм.	Лист	№ докум.	Подпись	Дат		

2. Разработка моделей данных.

Данный этап является самым важным при создании АИС. Здесь выделяются сущности, атрибуты сущностей и связи между сущностями. На основе полученной диаграммы “Сущность – связь” или логической модели строятся функциональные модели системы и диаграмма потоков данных. Для создания базы данных, нужно логическую модель представить в виде физической.

2.1 Концептуальная модель

Создание концептуальной модели базы данных начинается с тщательного анализа предметной области и выделения ключевых сущностей. В данном случае рассматривается система учёта посещений пациентов в поликлинике, что позволяет выделить несколько основных сущностей.

На основе вышеописанных сущностей и их атрибутов можно построить логическую модель базы данных, которая будет эффективно поддерживать процессы учёта и анализа в поликлинике.

Врач проводит прием в следствии которого пациенту ставится диагноз. Отсюда можно выделить следующие сущности - врач, прием и пациент. Для сущности врача будут выделены следующие атрибуты - id врача, фамилия, имя, отчество, специальность, стаж, а у сущности пациент - id пациента, фамилия, имя, отчество, снилс, дата рождения, адрес места проживания. Так как в ходе посещения пациенты определённого врача формируется сущность прием, которая будет содержать в себе id визита, id врача, id пациента, id диагноза, дату визита.

Далее представлен фрагмент концептуальной модели (Рис. 1)

					МИВУ 01.03.02	Лист
						7
Изм.	Лист	№ докум.	Подпись	Дат		

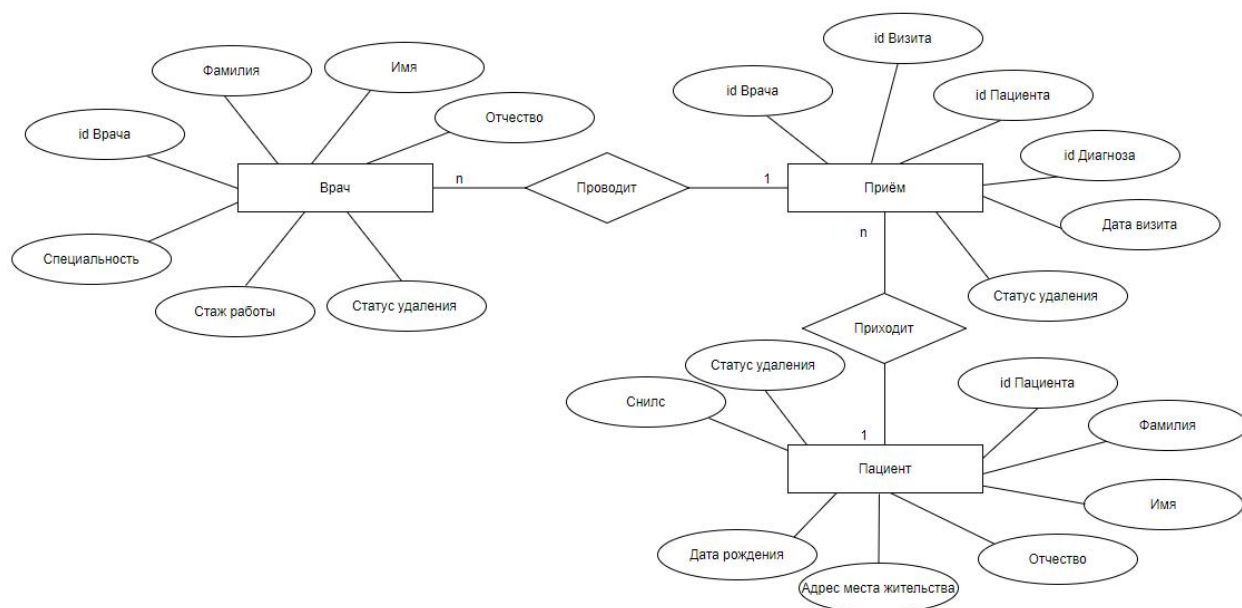


Рис. 1 – Связь между сущностями Врач, Прием и Пациент.

Кроме того, можно выделить связь между сущностями: один врач может принимать множество пациентов, а один пациент может посещать разных врачей. Это создаёт тип связи «многие ко многим», которая будет реализована через таблицу посещений.

В результате визита врач ставит пациенту диагноз. Следовательно, можно выделить сущность диагноз у которого будут атрибуты - id диагноза, название и описание диагноза.

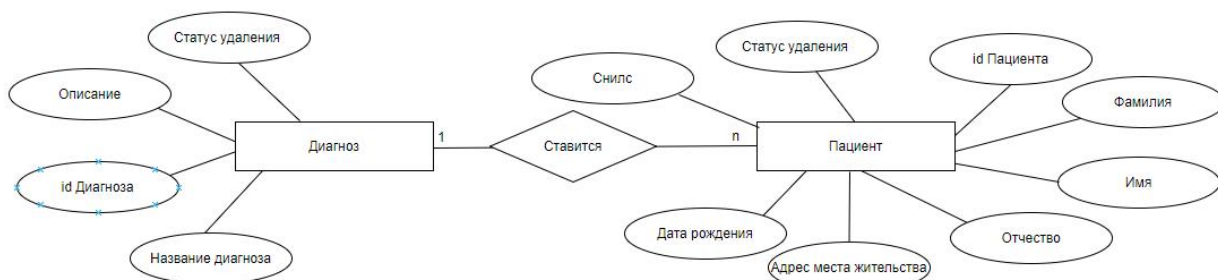


Рис. 2 – Связь между сущностями Пациент и Диагноз

Таким образом, для анализа заболеваемости и работы поликлиники необходимо создать логическую модель, которая позволит не только вести

учёт посещений, но и генерировать отчёты по количеству визитов, диагнозам и другим важным данным.

Полная схема концептуальной модели находится в приложении 1, рис. 1

2.2 Логическая модель

Логическая модель данных представляет собой абстрактное представление структуры данных, которое описывает, как данные будут организованы и взаимодействовать друг с другом в информационной системе. Она служит основой для проектирования базы данных и помогает определить, какие данные будут храниться, как они будут связаны и какие ограничения будут применяться.

Логическая модель данных является важным этапом в проектировании информационных систем. Она помогает структурировать данные, устанавливать связи между ними и обеспечивать целостность информации. Правильная разработка логической модели способствует созданию эффективной базы данных, которая будет отвечать требованиям бизнеса и обеспечивать надёжный доступ к данным для пользователей

Концептуальную модель можно преобразовывать по частям. Сущности врач, пациент, прием переведем в таблицы с следующими названиями: doctors, patients, visits. В таблицах doctors, patients, visits первичным ключом будут поля doctorid, patientid, visitid, по которым будет осуществлена связь с таблицей visits. Схема связи между тремя таблицами представлена ниже (Рис.4).

					МИВУ 01.03.02	Лист
						9
Изм.	Лист	№ докум.	Подпись	Дат		

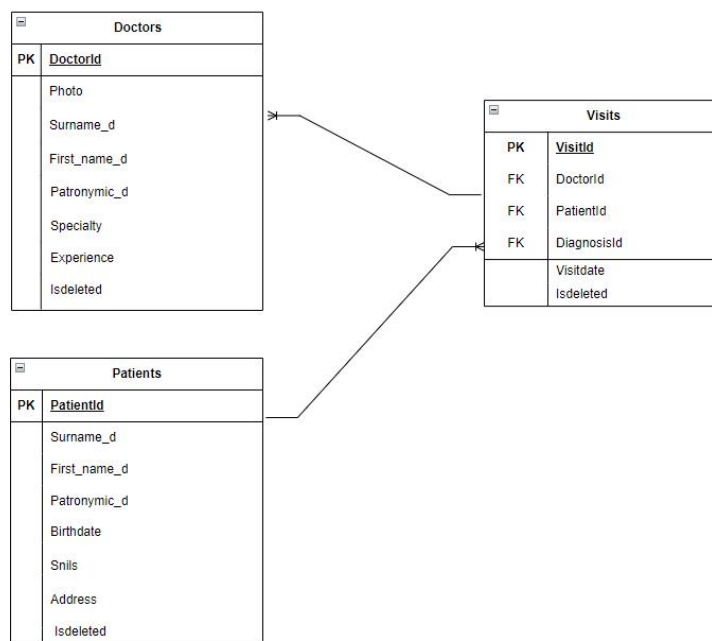


Рис. 3 - Таблицы Пациенты и Диагнозы

Далее рассмотрим сущность диагноз. В таблицу diagnoses первичным ключем будет поле diagnosisid. Сама таблица будет включать в себя поля diagnosisname, description, isdeleted.

Полная схема логической модели находится в приложении 1, рис. 2.

2.3 Физическая модель

Для создания базы данных будет использоваться СУБД Firebird 2.5.9.

Типы полей для таблиц представлены ниже (Рис.7)

Doctors	
PK	<u>DoctorId (integer)</u>
	Photo (blob)
	Surname_d (varchar(100))
	First_name_d (varchar(100))
	Patronymic_d (varchar(100))
	Specialty (varchar(50))
	Experience (integer)
	Isdeleted (smallint)

Рис. 4 – Таблица Doctors

Patients	
PK	<u>PatientId (integer)</u>
	Surname_d (varchar(100))
	First_name_d (varchar(100))
	Patronymic_d (varchar(100))
	Birthdate (date)
	Snils (varchar(20))
	Address (varchar(255))
	Isdeleted (smallint)

Рис. 5 – Таблица Patients

Visits	
PK	<u>VisitId (integer)</u>
FK	DoctorId (integer)
FK	PatientId (integer)
FK	DiagnosisId (integer)
	Visitdate (date)
	Isdeleted (smallint)

Рис. 6 – Таблица Visits

Diagnoses	
PK	<u>DiagnosisId (integer)</u>
	Diagnosisname (integer)
	Description (varchar(255))
	Isdeleted (smallint)

Рис. 7 – Таблица Diagnoses

Полная схема физической модели находится в приложении 1, рис. 3

3. Разработка и реализация ИС.

3.1 Руководство программиста

Приложение разработано в среде Visual Studio 2022 с использованием библиотеки FirebirdSql.Data.FirebirdClient для взаимодействия с базой данных Firebird. После установки библиотеки нужно подключить БД. Для этого используются следующие переменные:

```
FbConnectionStringBuilder fb_cons
```

```
FbConnection fbCon;
```

```
DataSet dsPatients;
```

```
DataSet dsDiagnoses;
```

```
FbDataAdapter adapter
```

Переменная FbConnectionStringBuilder fb_cons используется для построения строки соединения с базой данных Firebird. Здесь указываются следующие параметры:

```
DataSource = "localhost",
```

```
Port = 3050,
```

```
Role = "",
```

```
Dialect = 3,
```

```
Charset = "WIN1251",
```

```
UserID = "SYSDBA",
```

```
Password = "masterkey",
```

```
Database = @"C:\Users\Desktop\CLINICA.FDB"
```

Переменная FbConnection fbCon создана для соединение с базой данных Firebird. А переменная FbDataAdapter adapter используется для взаимодействия между приложением и базой данных Firebird. Используется для выполнения SQL-запросов (SELECT, INSERT, UPDATE, DELETE) и заполнения DataSet dsPatients, DataTable orglDataTable1, DataTable orglDataTable2 результатами запросов. DataSet dsPatients будет хранить полученные данные. Вывод данных будет совершаться на элементы DataGridView1, DataGridView2, которые позволяют отображать данные в табличной форме.

Основные запросы, которые использовались:

					МИВУ 01.03.02	Лист
						12
Изм.	Лист	№ докум.	Подпись	Дат		

- 1) SELECT * FROM doctors WHERE IsDeleted = 0
- 2) SELECT DISTINCT Specialty FROM Doctors
- 3) SELECT COUNT(*) FROM Doctors WHERE Surname_d = @Surname_d AND First_Name_d = @First_Name_d AND Patronymic_d = @Patronymic_d AND Specialty = @Specialty
- 4) INSERT INTO Doctors (Surname_d, First_Name_d, Patronymic_d, Specialty, Experience) VALUES (@Surname_d, @First_Name_d, @Patronymic_d, @Specialty, @Experience
- 5) SELECT Photo, Surname_d, First_Name_d, Patronymic_d, Specialty, Experience, ISDELETED FROM Doctors WHERE IsDeleted = 0 AND (UPPER(Surname_d) LIKE @InputText OR UPPER(First_Name_d) LIKE @InputText OR UPPER(Patronymic_d) LIKE @InputText OR UPPER(Specialty) LIKE @InputText)
- 5) UPDATE Doctors SET Surname_d = @Surname, First_Name_d = @FirstName, Patronymic_d = @Patronymic, Specialty = @Specialty, Experience = @Experience WHERE Surname_d = @OldSurname AND First_Name_d = @OldFirstName AND Patronymic_d = @OldPatronymic
- 6) UPDATE Doctors SET IsDeleted = 1 WHERE Surname_d = @Surname AND First_Name_d = @FirstName AND Patronymic_d = @Patronymic AND IsDeleted = 0
- 7) SELECT DoctorId, Surname_D AS Surname_d, First_Name_d AS First_Name_d, Patronymic_d AS Patronymic_d FROM Doctors WHERE IsDeleted = 0;
- 8) SELECT v.VisitId, d.Surname_D + ' ' + d.First_Name_d + ' ' + d.Patronymic_d AS DoctorFullName, p.Surname_D + ' ' + p.First_Name_d + ' ' + p.Patronymic_d AS PatientFullName, Snils, v.VisitDate, diag.DiagnosisName FROM Visits v JOIN Doctors d ON v.DoctorId = d.DoctorId JOIN Patients p ON v.PatientId = p.PatientId JOIN Diagnoses diag ON v.DiagnosisId = diag.DiagnosisId WHERE d.IsDeleted = 0 AND p.IsDeleted = 0 AND v.IsDeleted = 0

Подпрограммы:

- 1) LoadDoctors: загружает данные о врачах из базы данных Firebird. Использует FBDataAdapter для выполнения sql-запроса 'select'.
- 2) LoadPatients: загружает данные о пациентах из базы данных Firebird. Использует FBDataAdapter для выполнения sql-запроса 'select'.
- 3) LoadData: отображает данные о врачах и пациентах в базы данных Firebird в элементы управления DataGridView1 и DataGridView2

4) SoftDeleteDoctor: удаляет врача из базы данных, изменяя значение поля IsDeleted в соответствующей записи. Возвращает true при успехе и false при ошибке.

5) SoftDeletePatient: удаляет пациента из базы данных, изменяя значение поля IsDeleted в соответствующей записи. Возвращает true при успехе и false при ошибке.

6) SearchDoctors: поиск врачей в базе данных по переданной строке inputText.

7) SearchPatientss: поиск пациентов в базе данных по переданной строке inputText.

8) IsValidAddress: проверяет корректность адреса. Возвращает true или false.

Хранимые процедуры:

1) Для формирования отчёта перечень специалистов:

```
FOR SELECT d.Specialty, d.DoctorId, (d.Surname_d + ' ' + d.First_name_d + ' ' +  
d.Patronymic_d) AS FullName FROM Doctors d WHERE d.IsDeleted = 0 ORDER BY  
d.Specialty, FullName INTO :Specialty, :DoctorId, :FullName DO BEGIN SUSPEND; END
```

Триггеры:

```
1)AS BEGIN IF (NEW.DoctorID IS NULL) THEN NEW.DoctorID = NEXT VALUE FOR  
DoctorID_Seq; END
```

```
2)AS BEGIN IF (NEW.IsDeleted = 1) THEN BEGIN END END
```

Полный код разработанной программы находится в Приложении 2.

3.2 Руководство пользователя

Для организации работы приложения необходимо иметь версию Visual Studio 17.4.4 или старше.

При запуске программы открывается окно авторизации (Рис. 10)

Рис. 8 – Окно авторизации

После ввода логина и пароля откроется главное окно (Рис. 11)

Рис. 9 – Главное окно

В появившемся окне есть меню, благодаря которому осуществляется вызов другого окна (Forms 2): Диагнозы|Визиты.

Пользователь может получить отчёты (Рис.12) в целях анализа результатов работы поликлиники каждая кнопка предоставляет возможно автоматической генерации следующих документов:

- 1

—

T

4. Тестирование ИС.

При завершении разработки программы необходимо провести проверку на её работоспособность при различных действиях пользователей. Программа должна корректно отображать и выполнять действия пользователя в любом случае. В исключительных моментах должны выводиться окна с сообщениями, поясняющие ситуацию.

Проверим, как программа будет реагировать на некоторые действия пользователя. При нажатии на кнопку “удалить” перед пользователем открывается сообщение (Рис.1) о том, что действительно ли он хочет удалить данную запись. При подтверждении удаления записи выбранная строка удаляется из таблицы

Проверим, как работает добавление в базу (Рис.18-20)

Фото	Фамилия	Имя	Отчество	Специальность	Стаж (лет)
	Орлова	Милана	Ярославовна	Гастроэнтеролог	3
	Кириллов	Анатолий	Дмитриевич	Терапевт	5
	Макеева	Наталья	Леонидовна	Терапевт	5
	Иванов	Иван	Иванович	Хирург	6

Рис. 13 – Ввод данных в таблицу список врачей

Список врачей

Фото	Фамилия	Имя	Отчество	Специальность	Стаж (лет)
	Орлова	Милана	Ярославовна	Гастроэнтеролог	3
	Кириллов	Анатолий	Дмитриевич	Терапевт	5
	Макеева	Наталья	Леонидовна	Терапевт	5
	Иванов	Иван	Иванович	Хирург	6

Рис. 14, 15 – Сохранение данных

Допустим, пользователь введёт данные, которых нет в базе (Рис.21)

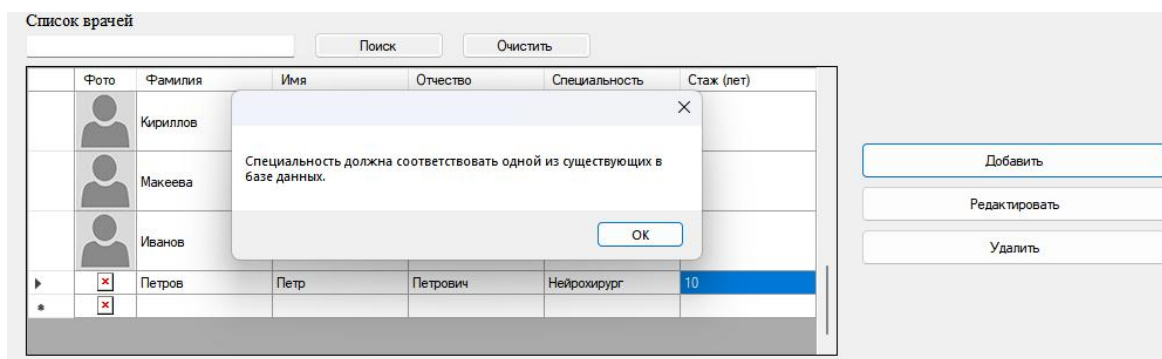


Рис. 16 – Сообщение об ошибке

Программа вывела сообщение об ошибке, не дав сохранить данные.

Допустим, пользователь введёт данные, которых будут использовать латинский алфавит (Рис.21)

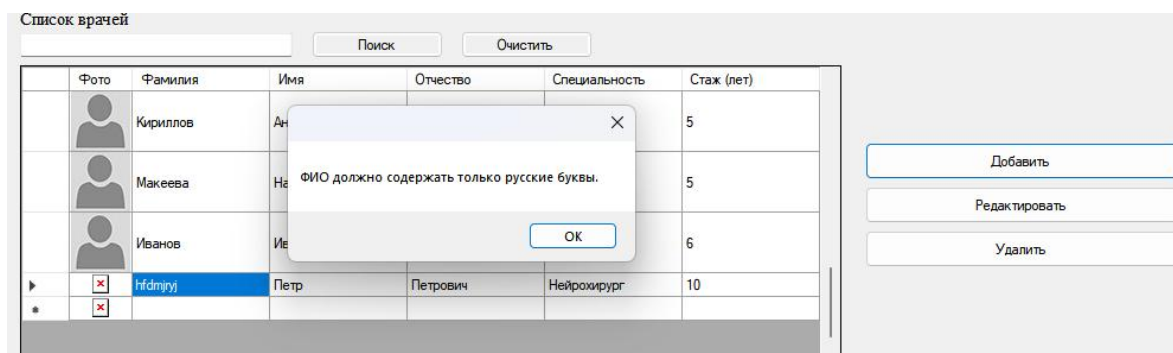


Рис. 17 – Сообщение об ошибке

Допустим, пользователь введёт данные, которых начинаться с строчной буквы, а не с заглавной (Рис.21)

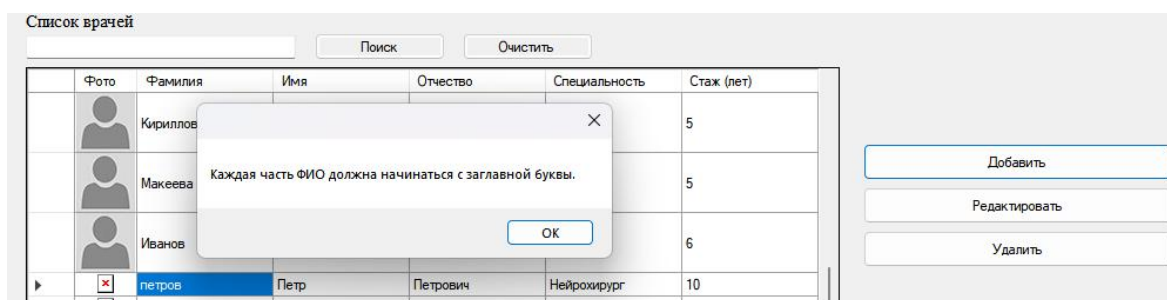


Рис. 18 – Сообщение об ошибке

Допустим, пользователь введёт данные, которых могут создать копию уже существующей записи (Рис.21)

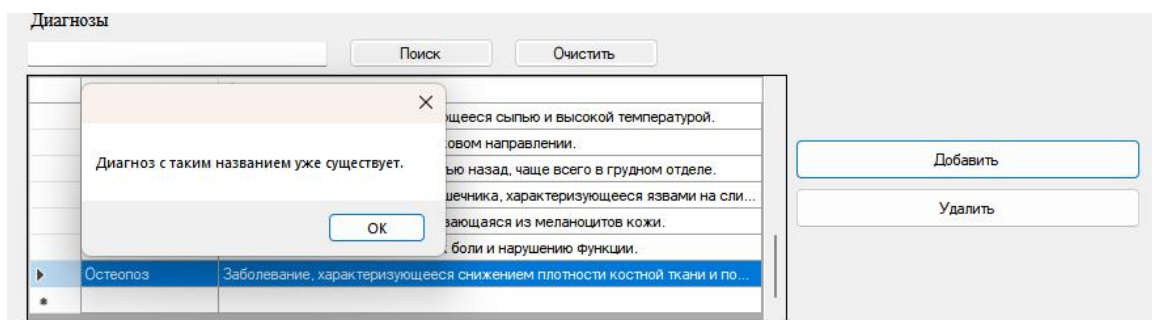


Рис. 19 – Сообщение об ошибке

Так же проверим, как работает поиск в таблице Пациенты. Допустим, пользователь ввёл фамилию пациента, которого нет в базе данных (Рис. 25)

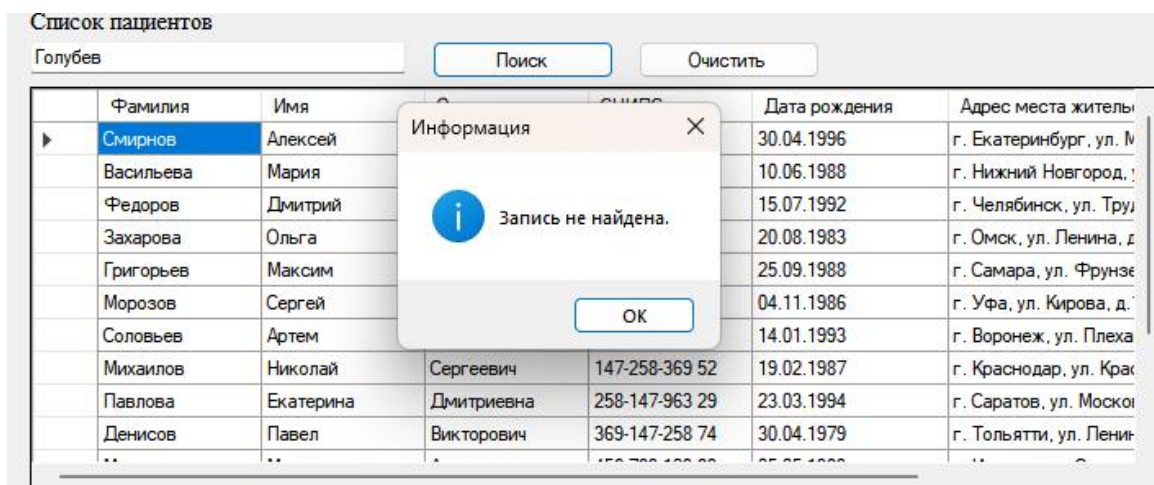


Рис. 20 – Сообщение об ошибке

Заключение.

В ходе данной курсовой работы была разработана автоматизированная информационная система для поликлиники. Для достижения поставленной цели была спроектирована структура БД, был разработан пошаговый алгоритм, на основе которого была создана программа, которая была реализована в среде программирования Visual Studio 2022 на языке программирования C# с использованием СУБД Firebird 2.5.9.

					МИВУ 01.03.02	Лист
						20
Изм.	Лист	№ докум.	Подпись	Дат		

Список используемой литературы.

1. Б. К. Татевосов. Основы SQL Татевосов, Б. К. (2018). Основы SQL: Учебное пособие. — М.: Инфра-М, 2018. — 240 с.
2. Н.Ф. Панова. Firebird. Установка, разработка баз данных, реализация запросов. Рекомендовано к изданию Редакционно-издательским советом федерального. ... П 16. Панова, Н.Ф. FireBird. Установка, разработка баз данных, реализация запросов: методические указания/Н.Ф. Панова; Оренбургский гос. ун-т.-Оренбург : ОГУ, 2014. - 45 с.
3. К. Хоффман. Проектирование баз данных. Хоффман, К. (2015). Проектирование баз данных: Практическое руководство по проектированию и реализации реляционных баз данных. — СПб.: БХВ-Петербург, 2015. — 320 с.
4. Базы данных. Часть 2: Практикум для студентов образовательных программ 09.03.02 Информационные системы и технологии; 09.03.03 Прикладная информатика; 01.03.02 Прикладная математика и информатика; 10.03.01 Информационная безопасность / сост. Фомин А.А. [Электронный ресурс]. – Электрон. текстовые дан. (1,5 Мб). - Муром: МИ ВлГУ, 2016. -1 электрон. опт. диск (CD-R). – Систем. требования: процессор x86 с тактовой частотой 500 МГц и выше; 512 Мб ОЗУ; Windows XP/7/8; видеокарта SVGA 1280x1024 High Color (32 bit); привод CD-ROM. - Загл. с экрана.
5. М. Дейт. Введение в базы данных. Дейт, М. Э. (2016). Введение в базы данных: Перевод с английского. — М.: Питер, 2016. — 720 с.

Приложение 1. Модели данных.

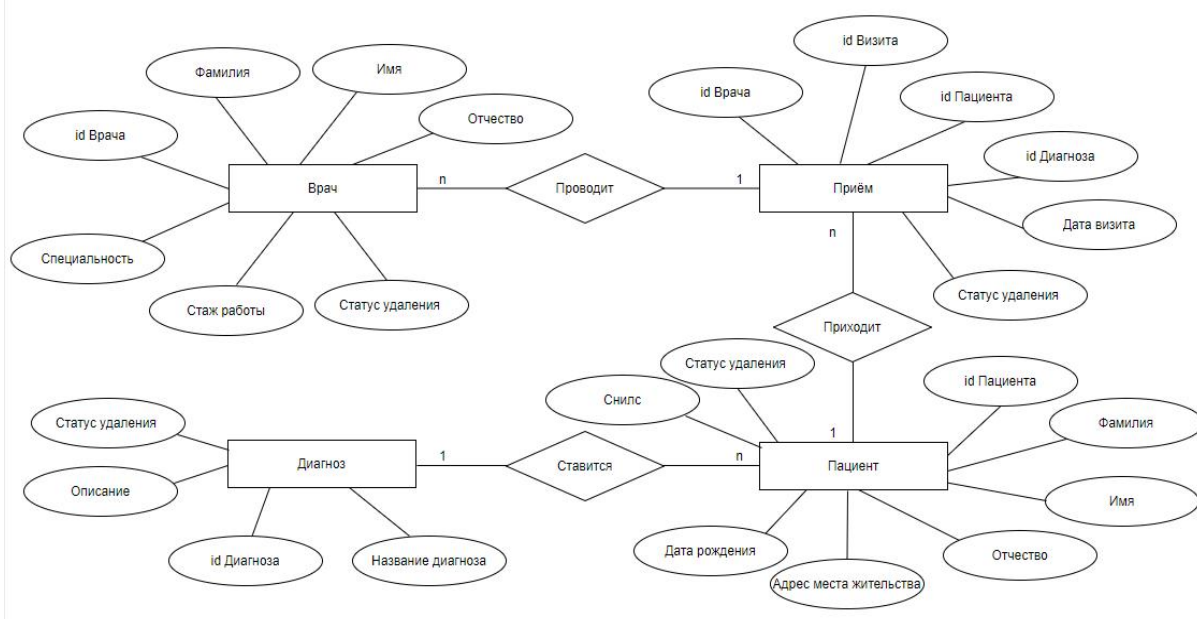


Рис. 1 – Концептуальная модель

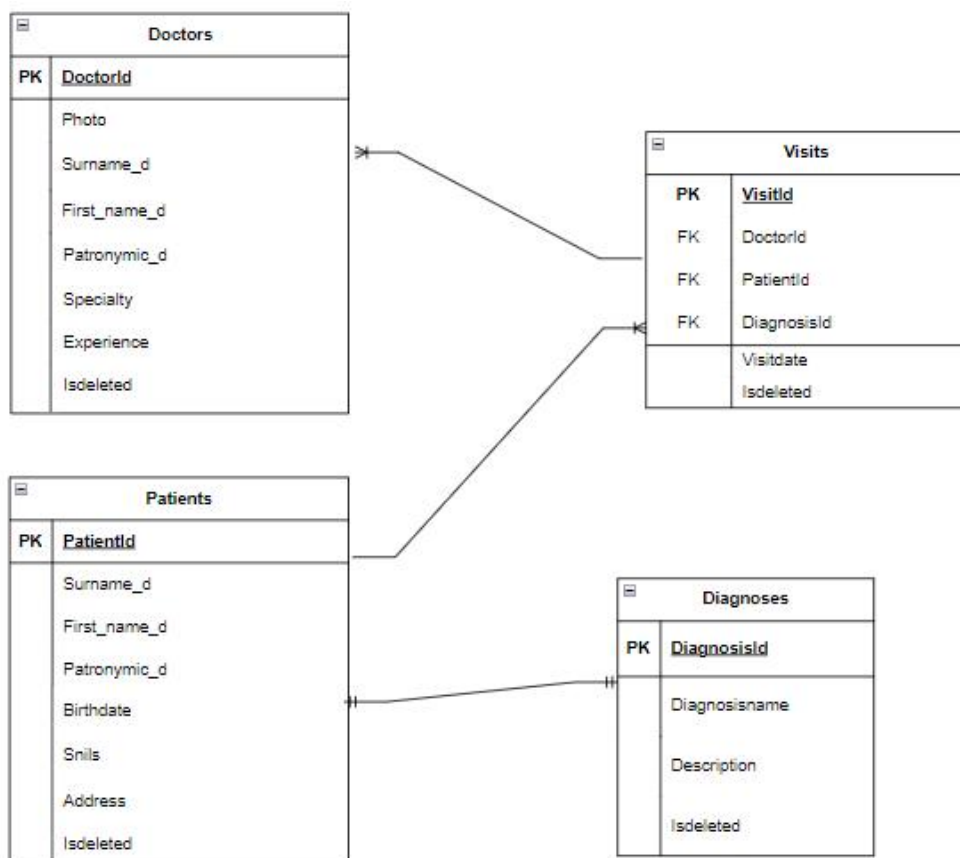


Рис. 2 – Логическая модель

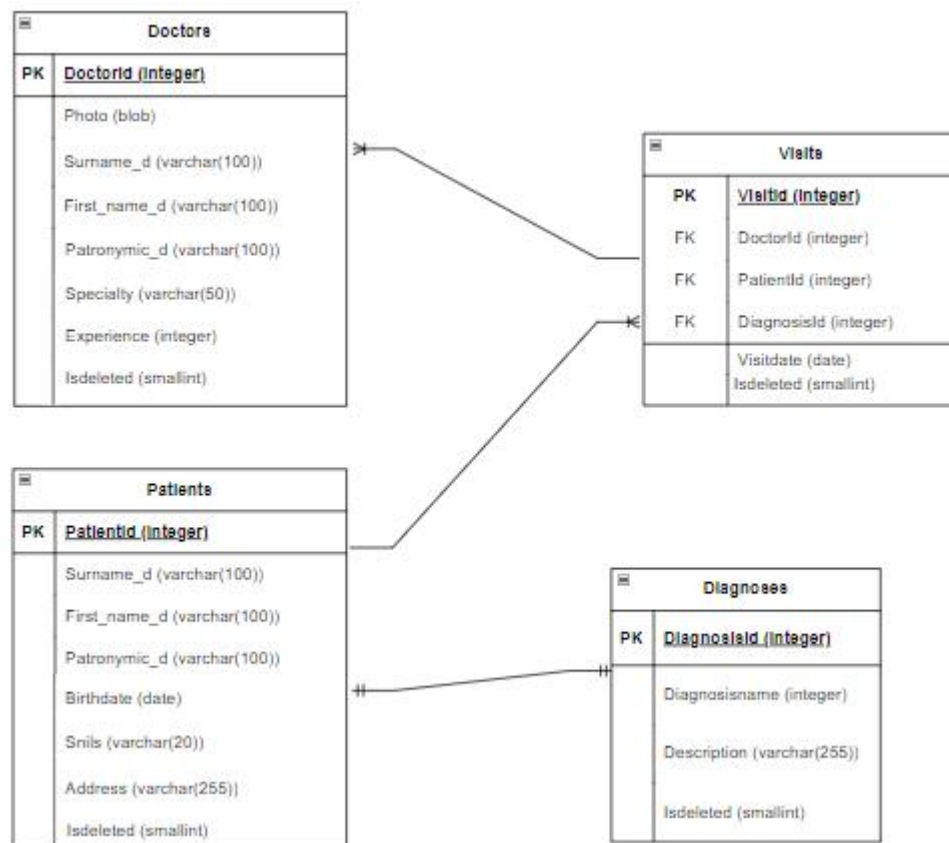


Рис. 3 – Физическая модель

Приложение 2. Текст программы.

Полный код представлен по ссылке: <https://github.com/ekkosolapova/123>

					МИВУ 01.03.02	Лист
						24
Изм.	Лист	№ докум.	Подпись	Дат		

Приложение 3.Снимки окон программы.

Авторизация

Логин

Пароль

Войти

Рис. 1 – Окно авторизации

Диагнозы | Визиты

Список врачей

Поиск Очистить

Фото	Фамилия	Имя	Отчество	Специальность	Стаж (лет)
	Кузнецов	Вадим	Анатольевич	Невролог	11
	Соловьев	Артём	Владимирович	Отоларинголог	5
	Ширяева	Лилия	Станиславовна	Психотерапевт	10
	Широкова	Ирина	Павловна	Кардиолог	4

Добавить

Редактировать

Удалить

Список пациентов

Поиск Очистить

Фамилия	Имя	Отчество	СНИЛС	Дата рождения	Адрес места жительства
Смирнов	Алексей	Валерьевич	135-792-468 84	30.04.1996	г. Екатеринбург, ул. И
Васильева	Мария	Николаевна	246-813-579 41	10.06.1988	г. Нижний Новгород, :
Федоров	Дмитрий	Олегович	369-258-149 86	15.07.1992	г. Челябинск, ул. Тру
Захарова	Ольга	Андреевна	159-753-864 36	20.08.1983	г. Омск, ул. Ленина, д
Григорьев	Максим	Викторович	951-753-852 34	25.09.1988	г. Самара, ул. Фрунзе
Морозов	Сергей	Владимирович	753-159-486 20	04.11.1986	г. Уфа, ул. Кирова, д.
Соловьев	Артём	Алексеевич	852-963-741 17	14.01.1993	г. Воронеж, ул. Плеха
Михайлов	Николай	Сергеевич	147-258-369 52	19.02.1987	г. Краснодар, ул. Крас
Павлова	Екатерина	Дмитриевна	258-147-963 29	23.03.1994	г. Саратов, ул. Моско
Денисов	Павел	Викторович	369-147-258 74	30.04.1979	г. Тольятти, ул. Лени

Добавить

Редактировать

Удалить

Рис. 2 – Главное окно программы

