Федеральное казенное профессиональное образовательное учреждение

«Новокузнецкий государственный гуманитарно-технический колледж-интернат»

Министерства Труда и социальной защиты Российской Федерации

Специальность: 09.02.04 «Информационные системы (по отраслям)»

КУРСОВОЙ ПРОЕКТ

по дисциплине: Средства диагностики и управления баз данных

на тему: Разработка приложения для автоматизации работы организации «Фитнес-клуб».

РАСЧЁТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Студент Скрипов Ю.А.

дата личная подпись расшифровка подписи

Руководитель проекта Грисман С.С.

дата личная подпись расшифровка подписи

Нормоконтроль Михайлов Н.Н,

дата личная подпись расшифровка подписи

Оценка

Новокузнецк, 2019 г.

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc27413247)

[1. ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА РАЗРАБОТКУ ПРОГРАММНОГО ПРОДУКТА 5](#_Toc27413248)

[ОБЩАЯ ЧАСТЬ 7](#_Toc27413249)

[2.1 Выбор и обоснование средств реализации БД 7](#_Toc27413250)

[2.2. Выбор и обоснование средств разработки программного продукта 9](#_Toc27413251)

[2.3 Анализ прототипов, существующих ИС 10](#_Toc27413252)

[3 СПЕЦИАЛЬНАЯ ЧАСТЬ 13](#_Toc27413253)

[3.1 Разработка алгоритма и проектирование структуры базы данных 13](#_Toc27413254)

[3.2 Разработка программного продукта 14](#_Toc27413255)

[3.3 Руководство пользователя 20](#_Toc27413256)

[ЗАКЛЮЧЕНИЕ 27](#_Toc27413257)

[СПИСОК ЛИТЕРАТУРЫ 28](#_Toc27413258)

# **ВВЕДЕНИЕ**

Компьютерные технологии вошли практически во все сферы деятельности человека, и это довольно легко объяснить: специализированные программные продукты способны облегчить ведение учета и хранения информации практически в любой отрасли. Владение информацией является важной составляющей успешного функционирования любой организации, в настоящее время информационные потоки увеличиваются, значит, повышаются требования к скорости обработки данных. Теперь уже большинство операций не может быть выполнено вручную, они требуют применение компьютерных технологий. Для обеспечения эффективности информационных процессов необходима соответствующая организация данных. С этой точки зрения информационные системы становятся подходящим способом долговременного хранения и обработки информации. Практически все экономические, финансовые и другие подобные системы используют соответствующую технологию в своей деятельности.

Целью данного курсового проекта является разработка информационной системы фитнес-клуба.

Актуальность темы обусловлена необходимостью создания приложения для облегчения и повышения эффективности работы администратора и клиентов фитнес-клуба.

Объектом в данной работе является «информационная система фитнес-клуба».

Предметом является возможность оптимизации процессов, связанных с обработкой информации и её обменом с пользователями, регистрация незарегистрированных, и авторизация в системе уже зарегистрированных клиентов, а также применение функций для коммуникабельности переходов в приложении.

При разработке курсового проекта были поставлены следующие задачи:

- Разработка технического задания;

- Проанализировать прототипы;

- Провести анализ и проектирование базы данных;

- Выбрать и обосновать программные средства реализации;

- Реализовать информационную систему;

- Провести тестирование и отладить программный продукт;

- Разработать руководство пользователя.

Разрабатываемое приложение сможет применяться как средство для регистрации, редактирования, удаления пользователей и услуг фитнес-клуба. Предоставления услуг зарегистрированным пользователям.

# **1. ТЕХНИЧЕСКОЕ ЗАДАНИЕ НА РАЗРАБОТКУ ПРОГРАММНОГО ПРОДУКТА**

Разработка приложения для автоматизации работы организации «Фитнес-клуб».

Прототипом данной системы можно считать интернет ресурс, имеющий функциональную часть работы данного приложения.

Требования к функциональным характеристикам следующие:

1. Необходимо предусмотреть два режима работы – режим администратора и режим пользователя.

2. В режиме пользователя должна быть возможность зарегистрироваться, заполнить свою анкету (ФИО, дата рождения, адрес, телефон, email). Реализовывать возможность смены пароля и редактирования данных.

3. В режиме пользователя можно оформить услугу, а также посмотреть историю своих услуг.

4. При оформлении услуги должна быть возможность выбора срока действия.

5. В режиме администратора должна быть возможность просматривать, добавлять и редактировать услуги. Каждая услуга характеризуется названием, сроком действия, стоимость, описанием, фото тренера.

6. В режиме администратора должна быть реализована возможность формирования услуги и регистрации пользователя в системе.

7. Также у администратора имеется возможность формировать отчеты: по клиенту, по услугам.

Требования к информационному обеспечению следующие:

К приложению должна быть предоставлена документация, содержащая техническое описание продукта и руководство пользователя.

Требования к лингвистическому обеспечению следующие:

Приложение должно иметь интерфейс и документацию на русском языке.

Требования к программному обеспечению следующие:

Разработка программного продукта предусматривается под OS Windows не ниже версии 7.

Требования к техническому обеспечению следующие:

Для работы приложения требуется MS SQL Server, Visual Studio С#. Также, необходимо не менее 2Гб оперативной памяти и 100Мб свободного места на жестком диске.

# **ОБЩАЯ ЧАСТЬ**

## **2.1 Выбор и обоснование средств реализации БД**

Основным требованием работы является наличие интерфейса приложения, разрабатываемого в СУБД MS SQL SERVER.

**База данных MS SQL SERVER** – это система, предназначенная для хранения и обработки информации. Комплекс таблиц, взаимосвязанных между собой, для доступа к которым применяется система управления базами данных (СУБД) MS SQL. MSSQL – это специальная программа с открытым кодом, которая используется на сервере SQL. Данная программа не способна обрабатывать большое количество информации, однако она идеальна для небольших и крупных баз данных.

В настоящее время, СУБД MS SQL является одной из самых известных, надежных и быстрых из всего семейства существующих СУБД.

MS SQL обладает большим количеством преимуществ перед другими системами:

1. СУБД MS SQL является одной из самых быстрых баз данных среди имеющихся на современном рынке.

2. СУБД MS SQL является высокопроизводительной и относительно простой в использовании СУБД, которую значительно проще инсталлировать и администрировать, чем многие другие большие системы.

3. СУБД MS SQL распространяется бесплатно для домашнего использования.

4. MS SQL понимает команды SQL, поддерживает интерфейс ODBC, протокол интерфейса с базами данных, разработанный компанией Microsoft.

5. Сервер позволяет подключаться одновременно неограниченному количеству пользователей. Доступ к серверу можно осуществить в интерактивном режиме с помощью различных интерфейсов, позволяющих вводить запросы и просматривать полученные результаты: это программы-клиенты, работающие с командной строкой, Web-броузеры, программы-клиенты, работающие в системе Windows, это, наконец, программные интерфейсы для языков C, Perl, Java, PHP и Python. Так что можно использовать как готовое клиентское программное обеспечение, так и создавать свое собственное.

Снижение экономических барьеров позволяет получить доступ к базам данных большему количеству людей и организаций, чем, когда бы то ни было.

Мощные реляционные СУБД могут теперь использовать как организации, так и индивидуальные пользователи для решения самых различных задач. И MS SQL – одна из таких систем.

При написании приложения был выбран язык программирования C#, так как именно на серверах MS SQL получил широкое распространение в настоящее время. C# является одним из лучших языков программирования, которые подходят для написания приложений.

Таким образом основными критериями выбора данной СУБД были:

* Многопоточность. Поддержка нескольких одновременных запросов;
* Оптимизация связей с присоединением многих данных за один проход;
* Записи фиксированной и переменной длины;
* Гибкая система привилегий и паролей;
* Основанная на потоках, быстрая система памяти;
* Утилита проверки и ремонта таблицы (isamchk);
* Все операции работы со строками не обращают внимания на регистр символов в обрабатываемых строках;
* Псевдонимы применимы как к таблицам, так и к отдельным колонкам в таблице;
* Все поля имеют значение по умолчанию. INSERT можно использовать на любом подмножестве полей;
* Легкость управления таблицей, включая добавление и удаление ключей и полей.

Итак, популярность MS SQL сервера определяется не только тем, что пакет распространяется бесплатно и имеет высокую производительность, а также тем, что у него есть множество тех достоинств, которые не имеют другие СУБД.

В разрабатываемом веб-приложении необходимо добавлять, хранить, удалять и обрабатывать большой объём информации, поэтому выгоднее использовать MS SQL, потому что он, кроме своих преимуществ, предоставляет эту возможность совершенно бесплатно. Это действительно одна из самых мощных, быстрых и надежных СУБД.

## **2.2. Выбор и обоснование средств разработки программного продукта**

При проектировании приложения были использованы такие программные средства как Microsoft Visual Studio.

Выбор редактора. Редактор исходного кода — текстовый редактор для создания и редактирования исходного кода программ. Он может быть отдельным приложением, или встроен в интегрированную среду разработки. Таких редакторов существует огромное количество. Большинство из них имеет следующие удобные возможности:

* подсветка синтаксиса;
* автодополнение;
* отступы;
* проверка правильности расстановки скобок;
* контекстная помощь по коду.

При реализации программного продукта были проанализированы редактор как Microsoft Visual Studio.

Microsoft Visual Studio — редактор исходного кода, разработанный Microsoft для Windows, Linux и macOS. Позиционируется как «лёгкий» редактор кода для кроссплатформенной разработки веб- и облачных приложений. Включает в себя отладчик, инструменты для работы с Git, подсветку синтаксиса, IntelliSenseи средства для рефакторинга. Имеет широкие возможности для кастомизации: пользовательские темы, сочетания клавиш и файлы конфигурации. Распространяется бесплатно, разрабатывается как программное обеспечение с открытым исходным кодом, но готовые сборки распространяются под проприетарной лицензией.

## **2.3 Анализ прототипов, существующих ИС**

При реализации программного продукта, разработчик анализирует и систематизирует данные, просматривая аналогичные приложения. На данном этапе происходит весь сбор информации, выявление достоинств и недостатков, которые имеются у прототипов. После проведённой работы, он продумывает план создания приложения, куда будут включены плюсы, а также исключены ошибки прототипов.

Схема анализа прототипов включает в себя:

* дизайн приложения и общее впечатление от него;
* доступность контактной информации для пользователей;
* структура программного продукта и удобство использования;
* навигация в приложении;
* отображение в разных web–браузерах;
* объем и актуальность представленной информации в данном программном продукте;
* удобство использования панели администратора данного приложения.

Разработка приложения для автоматизации работы организации «Фитнес-клуб».

Прототипом данной системы можно считать интернет ресурс, имеющий функциональную часть работы данного приложения.

Интернет ресурс «Formula-fitness», на которое можно перейти по ссылке https://apps.apple.com/us/app/formula-fitness/id1349118438?l=ru&ls=1 представлено на рисунке 1, 2, 3.

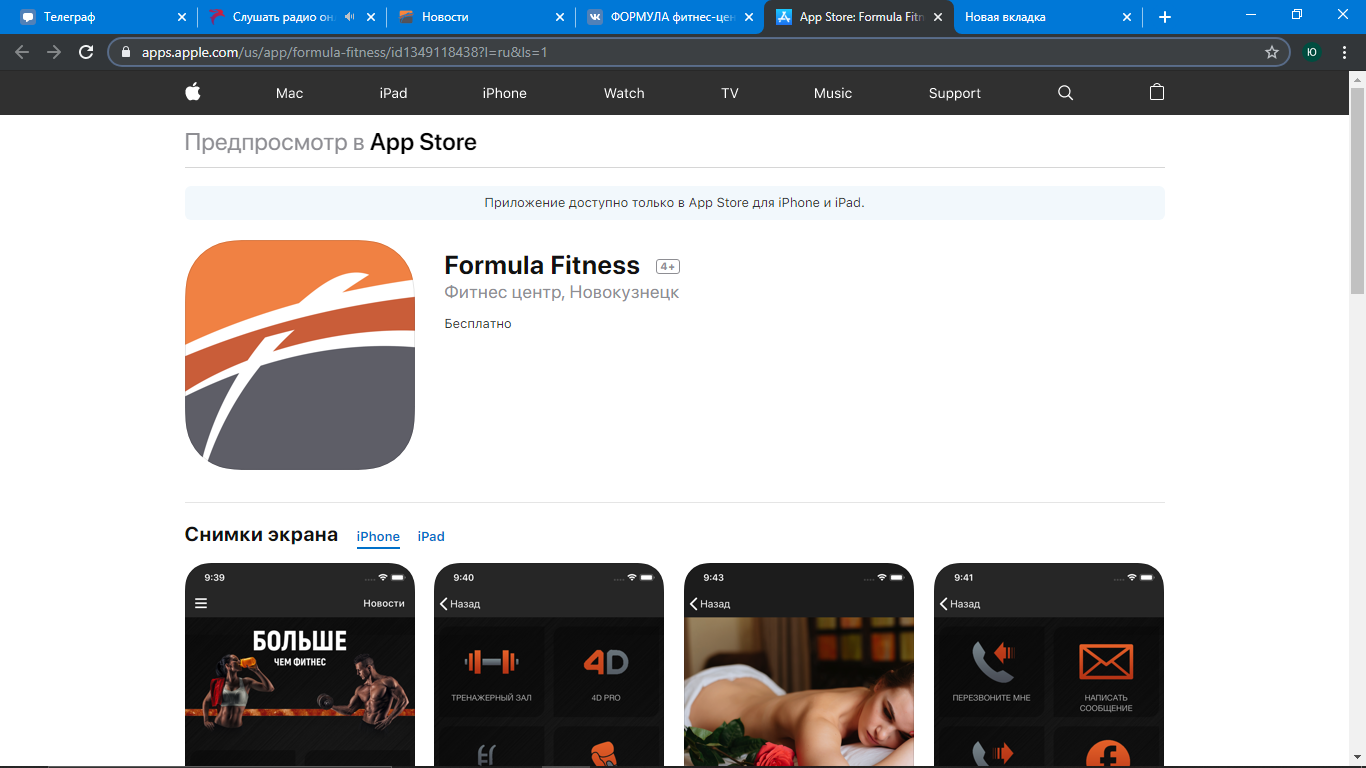


Рисунок 1 - «Fitness-Formula»

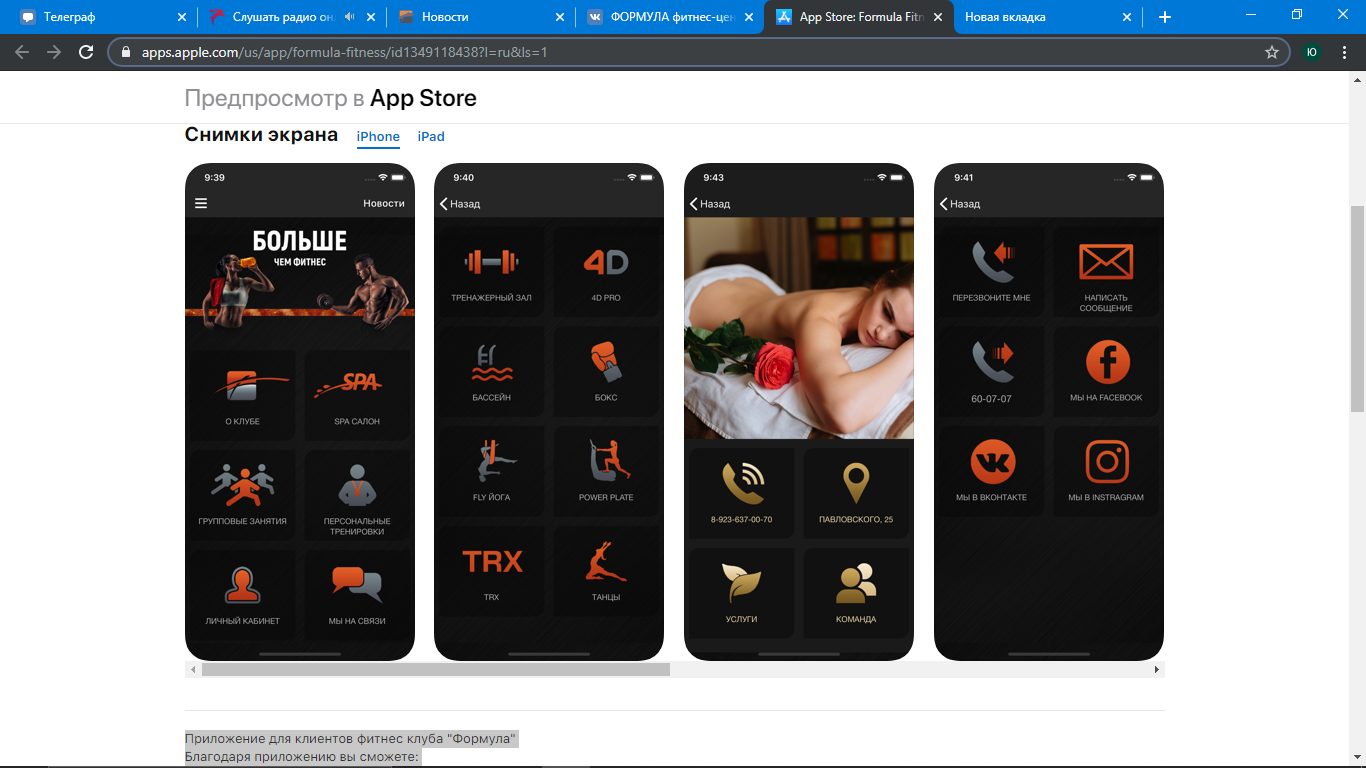


Рисунок 2 – «Приложение Fitness-formula»

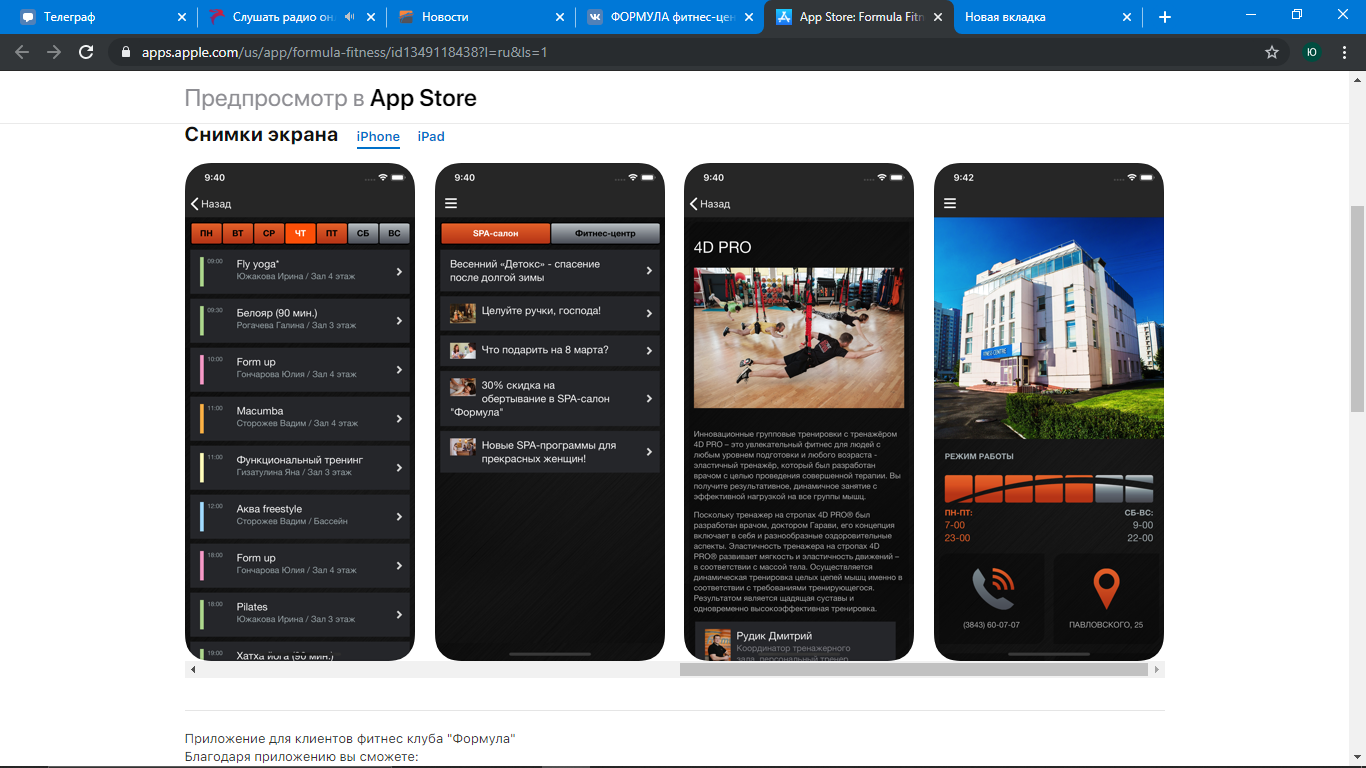


Рисунок 3 – «Приложение фитнес клуба «Формула»»

Благодаря приложению вы сможете:

- получить полную информацию об услугах фитнес клуба и spa-салона;

- посмотреть актуальное расписание групповых занятий;

- узнать виды персональных тренировок;

- написать сообщение администрации клуба;

- получать уведомления об изменениях в работе клуба;

- ознакомиться со специальными предложениями клуба и узнать о предстоящих мероприятиях.

# **3 СПЕЦИАЛЬНАЯ ЧАСТЬ**

## **3.1 Разработка алгоритма и проектирование структуры базы данных**

База данных «Fitness-Club» была обоснована на основании задания, представленного в приложении А.

Модель базы данных «Fitness-Club» создана в MS SQL SERVER Management Studio 18.

При создании таблиц базы данных использовался режим конструктора MS SQL, где были установлены свойства полей всех таблиц данной базы данных.

База данных «Fitness-Club» состоит из четырех таблиц:

Для создания таблиц использовался режим конструктора, позволяющий установить свойства полей таблицы.

Модель базы данных «Fitness-Club» состоит из четырех таблиц, каждая из которых имеет свою структуру, необходимую для установления связи между всеми таблицами:

* Users (Пользователи);
* Instructors (Инструкторы);
* Uslugi (Услуги);
* Service (Сервис).

Таблица «Users» предназначена для входа в систему в режиме администратора и пользователя при вводе логина и пароля. Имеет связь с таблицей Service (сервис). Ключевым полем является ID\_Users (Код пользователя), тип данных int. Также имеютcя поля FIO (ФИО), тип данных nchar (50). Email (электронная почта), тип nchar (50). Phone (Телефон), тип nchar (50). Address (адрес), тип nchar (50). type (класс), тип bit.

Таблица Instructors (Инструкторы) имеет связь с таблицей Service (сервис). Ключевым полем является поле ID\_Instructor (код инструктора), тип данных int. Name\_Instructor (имя инструктора), тип данных nchar (50). Phone (телефон), тип данных nchar (50). Photo (фото), тип данных nchar (50).

Таблица Uslugi (услуги) имеет связь с таблицей Service (сервис). Ключевым полем является поле ID\_Uslugi (код услуги), тип данных int. Name\_Uslugi (имя услуги), тип данных nchar (50).

Таблица Service (сервис). Ключевым полем является ID\_Service (код сервиса), тип данных int. ID\_Uslugi (код услуги), тип данных int, поэтому полю имеет связь с таблицей Uslugi. ID\_Users (код пользователя), тип данных int, это поле связывает с таблицей Users. ID\_Instructor (код инструктора), тип данных int, это поле связывает с таблицей Instructors. Price (цена), тип данных nchar(50). Start\_Time (время начала), тип данных nchar(50). End\_Time (время окончания), тип данных nchar(50).

Целостность данных – это значит, что данные будут надёжно хранится в базе данных. СУБД автоматически поддерживает целостность данных с помощью установленных связей между таблицами. Таблицы базы данных связаны связями «один ко многим»; для каждой из связей установлена поддержка целостности данных, каскадные обновление и удаление. Схема связей таблиц представлена в соответствии с рисунком 3.

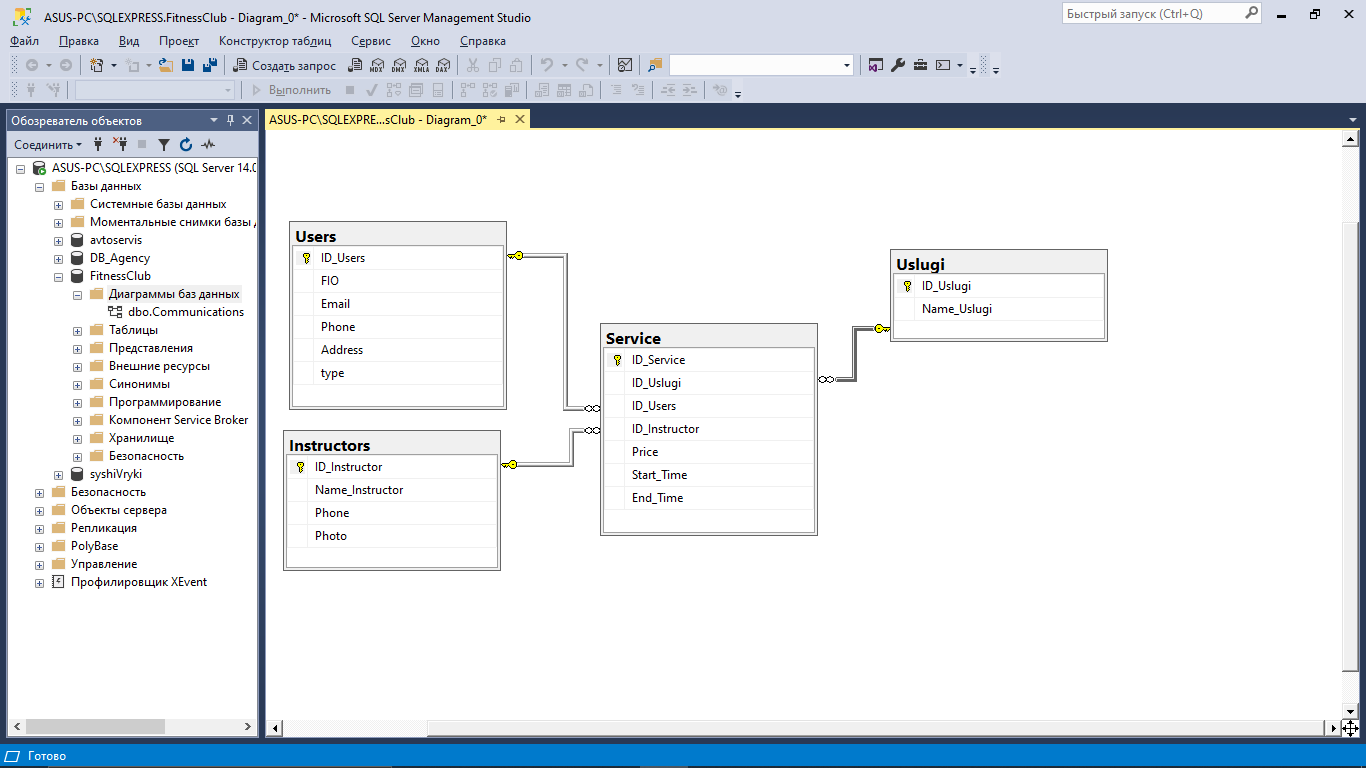


Рисунок 4 – «Схема связей таблиц базы данных»

## **3.2 Разработка программного продукта**

При разработке данного программного продукта было реализовано несколько подсистем. Первая из них – система авторизации. В ней реализован механизм, где работа происходит с зарегистрированными пользователями. Алгоритм работы выглядит так, что при загрузке приложения, происходит проверка на регистрацию пользователя в базе программы. Если она имеется, то авторизация не предлагается, а в противном случае, необходимо её пройти. Данный механизм реализован посредством простой проверкой существования записи в таблице. Проверка идет SQL запросом в присоединенную к программе базу данных. Код данной части программного продукта представлен на листинге 1.

Листинг 1. Проверка пользователя

string s = "select \* from Users where FIO='" + FIO + "'and Email='" + Email + "'";

В данном действии, как и во всей системе происходит работа с базой данных. После ввода логина и пароля происходит отправка запроса на выборку имени пользователя (FIO) и пароля (Email) после возвращает результат выполнения. В программе предусмотрено предупреждение в случае ввода ошибочных данных. Представлено на листинге 2.

Листинг 2

else

{

MessageBox.Show("Пользователь не найден");

}

При заполнении необходимых полей происходит регистрация нового пользователя в базу данных, код представлен на листинге 3.

Листинг 3. Регистрация пользователя

string s = "insert into Users" +

"(FIO,Email,Phone,Address,type) Values" +

"('" + textBox1.Text.Trim() + "'," +

"' " + textBox2.Text.Trim() + "'," +

"' " + maskedTextBox1.Text.Trim() + "'," +

"' " + textBox4.Text.Trim() + "', 0)";

SqlCommand comm = new SqlCommand(s, conn);

comm.ExecuteScalar();

Если авторизуемый пользователь «администратор», то открывается форма администратора, если «пользователь» - форма пользователя. Происходит проверка на тип пользователя.

Лизинг 4. Проверка на тип пользователя

reader.Read();

Program.ID\_Users = reader.GetInt32(0);

Program.type\_user = reader.GetBoolean(5);

if (Program.type\_user == false)

{

Form2 f = new Form2();

f.textBox1.Text = reader.GetString(1);

f.textBox2.Text = reader.GetString(2);

f.maskedTextBox1.Text = reader.GetString(3);

f.textBox4.Text = reader.GetString(4);

f.Show();

}

else

{

Form3 f = new Form3();

f.Show();

Вторая подсистема – управление пользователями. Она отвечает за открытие форм добавления, удаления, редактирование пользователей.

Коды программ указаны на лизингах 5, 6, 7.

Лизинг 5. Добавление пользователя.

if(textBox1.Text.Trim()==""||

textBox2.Text.Trim()==""||

maskedTextBox1.Text.Trim()==""||

textBox4.Text.Trim() == "")

{

MessageBox.Show("Не все поля заплненны!");

}

else

{

SqlConnection conn = new SqlConnection(Program.st\_connect);

conn.Open();

string s1 = "Select \* from Users where FIO='" + textBox1.Text.Trim() + "'";

SqlCommand comm1 = new SqlCommand(s1, conn);

SqlDataReader read1 = comm1.ExecuteReader();

if (read1.HasRows)

{

MessageBox.Show("Логин занят!");

}

else

{

read1.Close();

string s = "insert into Users" +

"(FIO,Email,Phone,Address,type) Values" +

"('" + textBox1.Text.Trim() + "'," +

"' " + textBox2.Text.Trim() + "'," +

"' " + maskedTextBox1.Text.Trim() + "'," +

"' " + textBox4.Text.Trim() + "', 0)";

SqlCommand comm = new SqlCommand(s, conn);

comm.ExecuteScalar();

this.Close();

}

conn.Close();

}

}

Лизинг 6. Удаление пользователя

int k = dataGridView1.CurrentRow.Index;

Program.ID\_Users = Convert.ToInt32(dataGridView1[0, k].Value);

SqlConnection conn = new SqlConnection(Program.st\_connect);

conn.Open();

string s = "delete from Users where ID\_Users= " + Program.ID\_Users.ToString();

SqlCommand comm = new SqlCommand(s, conn);

comm.ExecuteScalar();

conn.Close();

MessageBox.Show("Клиент удалён!");

this.Close();

Form3 f3 = new Form3();

f3.Show();

Лизинг 7. Редактирование пользователя

private void button1\_Click(object sender, EventArgs e)

{

button1.Visible = false;

button2.Visible = true;

textBox1.Enabled = true;

textBox2.Enabled = true;

maskedTextBox1.Enabled = true;

textBox4.Enabled = true;

}

private void button2\_Click(object sender, EventArgs e)

{

SqlConnection connect = new SqlConnection(Program.st\_connect);

connect.Open();

string s = "update Users set " +

"FIO='" + textBox1.Text.Trim() + "', "

+ "Email='" + textBox2.Text.Trim() + "', "

+ "Phone='" + maskedTextBox1.Text.Trim() + "', "

+ "Address='" + textBox4.Text.Trim() + "' "

+ "where ID\_Users=" + Program.ID\_Users.ToString();

SqlCommand comm = new SqlCommand(s, connect);

comm.ExecuteScalar();

connect.Close();

button2.Visible = false;

button1.Visible = true;

textBox1.Enabled = false;

textBox2.Enabled = false;

maskedTextBox1.Enabled = false;

textBox4.Enabled = false;

}

Следующая подсистема отвечает за добавление, удаление инструкторов и услуг фитнесс-клуба. Код представлен на лизингах

Лизинг 8. «Добавление инструкторов»

if (textBox1.Text.Trim() == "" ||

maskedTextBox1.Text.Trim() == "")

{

MessageBox.Show("Не все поля заполненны!");

}

else

{

SqlConnection conn

= new SqlConnection(Program.st\_connect);

conn.Open();

string s1 = "select \* from Instructors where Name\_instructor='" +

textBox1.Text.Trim() + "'";

SqlCommand comm1 = new SqlCommand(s1, conn);

SqlDataReader read12 = comm1.ExecuteReader();

if(read12.HasRows)

{

MessageBox.Show("Логин занят!");

}

else

{

read12.Close();

string s = "insert into Instructors " + "" +

"(Name\_Instructor,Phone,Photo) values " +

"('" + textBox1.Text.Trim() + "'," +

" '" + maskedTextBox1.Text.Trim() + "'," +

"'" + label3.Text + "')";

SqlCommand comm = new SqlCommand(s, conn);

comm.ExecuteScalar();

conn.Close();

Лизинг 9. «Удаление инструкторов»

int k = dataGridView1.CurrentRow.Index;

int ID\_Instructor = Convert.ToInt32(dataGridView1[0, k].Value);

SqlConnection conn = new SqlConnection(Program.st\_connect);

conn.Open();

string s = "DELETE FROM Instructors WHERE ID\_Instructor=" + ID\_Instructor.ToString();

SqlCommand comm = new SqlCommand(s, conn);

comm.ExecuteScalar();

conn.Close();

this.instructorsTableAdapter.Fill(this.fitnessClubDataSet5.Instructors);

Лизинг 10. «Добавление услуги»

SqlConnection conn

= new SqlConnection(Program.st\_connect);

conn.Open();

string s1 = "select \* from Uslugi where Name\_Uslugi='" +

textBox1.Text.Trim() + "'";

SqlCommand comm1 = new SqlCommand(s1, conn);

SqlDataReader read12 = comm1.ExecuteReader();

if (read12.HasRows)

{

MessageBox.Show("Логин занят!");

}

else

{

read12.Close();

string s = "insert into Uslugi " + "" +

"(Name\_Uslugi) values " +

"('" + textBox1.Text.Trim() + "')";

SqlCommand comm = new SqlCommand(s, conn);

comm.ExecuteScalar();

conn.Close();

Form8 f = new Form8();

f.Show();

this.Close();

Лизинг 11. «Удаление услуги»

int k = dataGridView1.CurrentRow.Index;

int ID\_Uslugi = Convert.ToInt32(dataGridView1[0, k].Value);

SqlConnection conn = new SqlConnection(Program.st\_connect);

conn.Open();

string s = "DELETE FROM Uslugi WHERE ID\_Uslugi=" + ID\_Uslugi.ToString();

SqlCommand comm = new SqlCommand(s, conn);

comm.ExecuteScalar();

conn.Close();

this.uslugiTableAdapter.Fill(this.fitnessClubDataSet6.Uslugi);

Следующая подсистема – создание отчетов. В ней можно создать отчет по клиентам и услугам фитнесс-клуба.

Лизинг 12. Отчет по клиентам

Microsoft.Office.Interop.Excel.Application myExcel = new Microsoft.Office.Interop.Excel.Application();//создаем виртуальный объект Excel

//создать книгу в объекте Excel

myExcel.Application.Workbooks.Add(Type.Missing);

//Настраиваем ячейки

myExcel.Columns.ColumnWidth = 15;

//Пишем заголовки ячеек

for (int i = 0; i < dataGridView1.ColumnCount; i++)

{

myExcel.Cells[1, i + 1] = dataGridView1.Columns[i].HeaderText;// заголовки программно

}

/\*myExcel.Cells[1, 2] = "Заголовок 2"; //заголовки прописаны

myExcel.Cells[1, 3] = "Заголовок 3";

myExcel.Cells[1, 4] = "Заголовок 4";\*/

for (int i = 0; i < dataGridView1.ColumnCount; i++)

{

for (int j = 0; j < dataGridView1.RowCount - 1; j++)

{

myExcel.Cells[j + 2, i + 1] = dataGridView1[i, j].Value.ToString();

}

}

Range range = (Range)myExcel.Columns[1, Type.Missing];

range.EntireColumn.Hidden = true;

myExcel.Visible = true;

Лизинг 13. Отчет по услугам

Microsoft.Office.Interop.Excel.Application myExcel = new Microsoft.Office.Interop.Excel.Application();

myExcel.Application.Workbooks.Add(Type.Missing);

myExcel.Columns.ColumnWidth = 15;

for (int i = 0; i < dataGridView1.ColumnCount; i++)

{

myExcel.Cells[1, i + 1] = dataGridView1.Columns[i].HeaderText

}

for (int i = 0; i < dataGridView1.ColumnCount; i++)

{

for (int j = 0; j < dataGridView1.RowCount - 1; j++)

{

myExcel.Cells[j + 2, i + 1] = dataGridView1[i, j].Value.ToString();

}

}

Range range = (Range)myExcel.Columns[1, Type.Missing];

range.EntireColumn.Hidden = true;

myExcel.Visible = true;

# **3.3 Руководство пользователя**

Для того, чтобы зайти в данное приложение, необходимо запустить C#, открыть проект «FitnessClub».

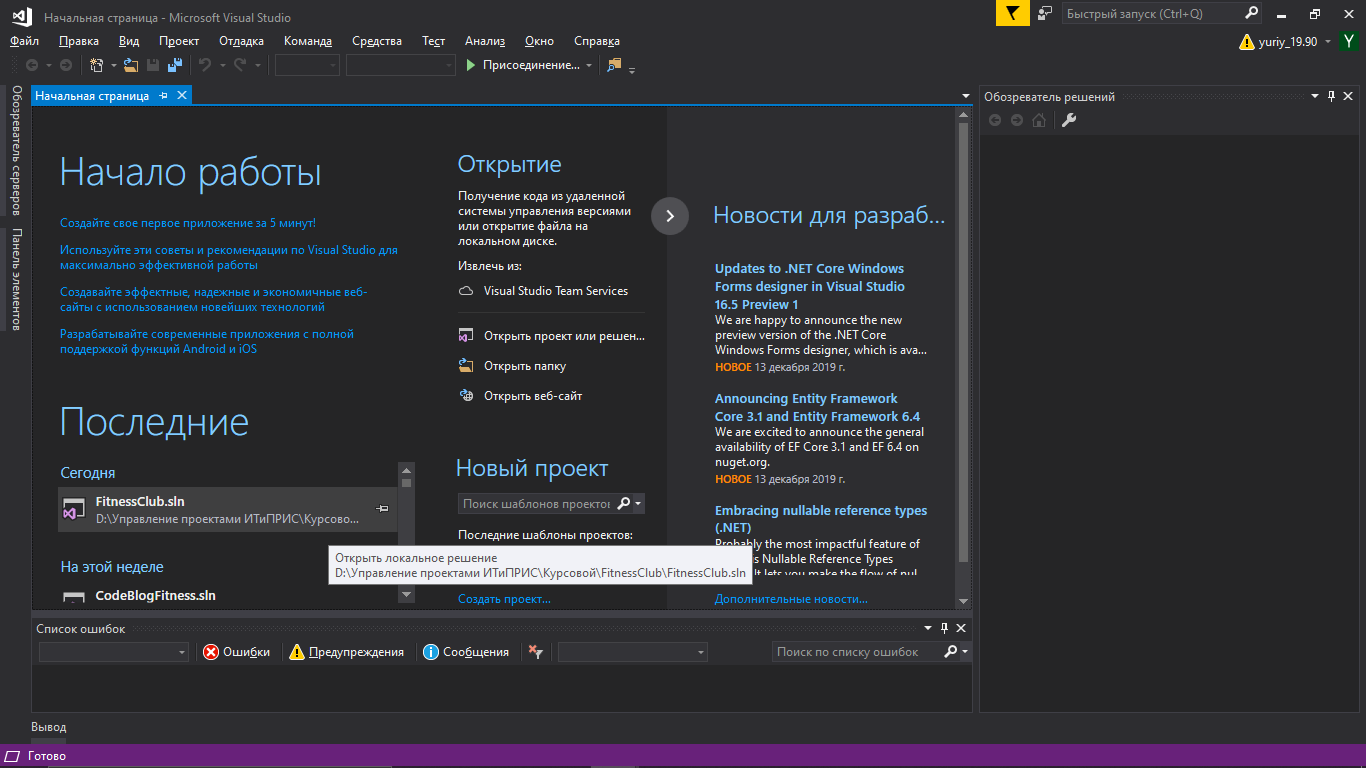


Рисунок 5 – «Запуск программы»

После этого запускается стартовая форма проекта «Авторизация». В ней предлагается войти в систему уже зарегистрированным клиентам или зарегистрироваться в ней.

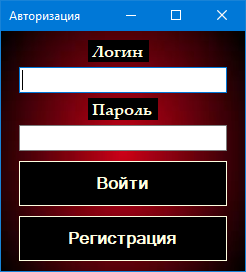


Рисунок 6 – «Форма авторизации»

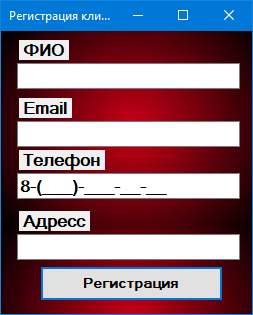


Рисунок 7 – «Форма регистрации»

Клиенты переходят на форму «Клиент». Где могут редактировать свои данные и выбрать услуги фитнесс-клуба.

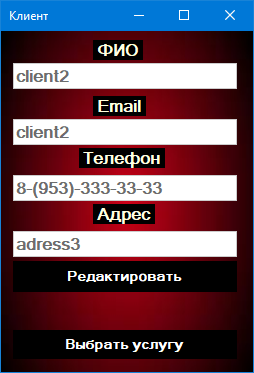


Рисунок 8 – «Форма «Клиент»»

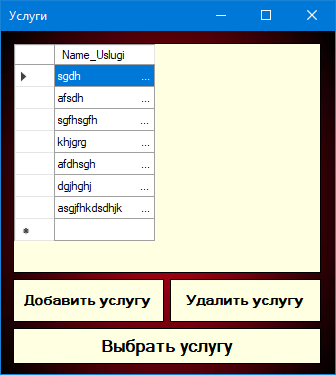


Рисунок 9 – «Форма «Услуги»

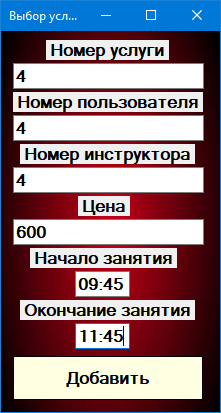


Рисунок 10 – «Форма выбора услуги»

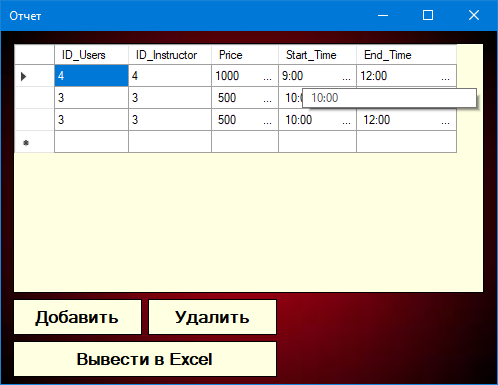


Рисунок 11 – «Форма отчета по услугам»

Администраторы в этой системе могут регистрировать, редактировать клиентов. Добавлять, удалять инструкторов и услуги. А также делать отчеты по услугам и клиентам фитнесс-клуба.

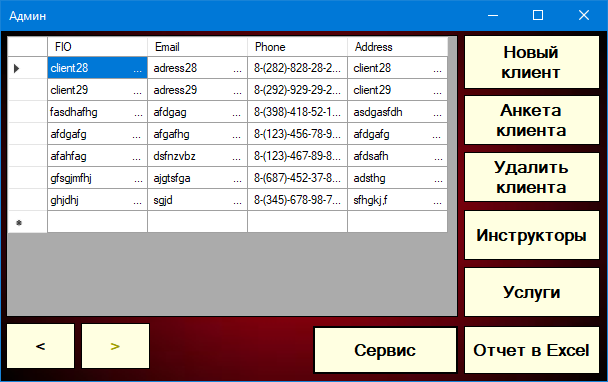


Рисунок 12 – «форма администратора»

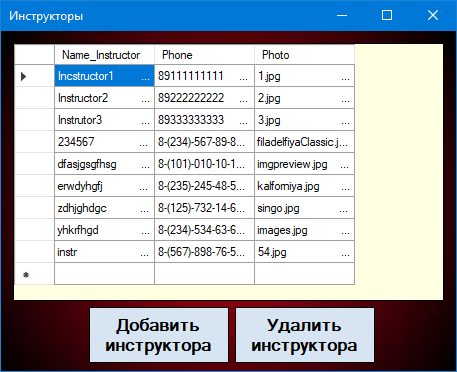


Рисунок 13 – «Форма «Инструкторы»»

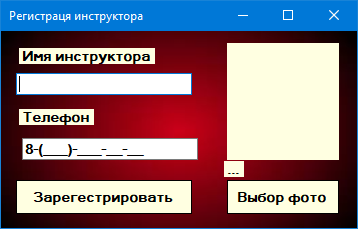


Рисунок 14 – «Форма регистрации инструктора»

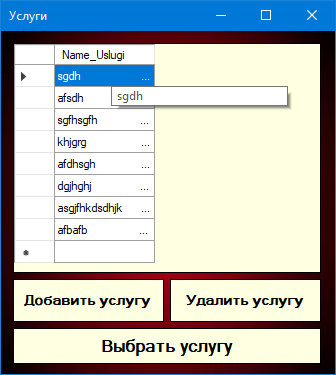


Рисунок 15 – «Форма добавления и удаления услуг»

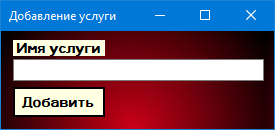


Рисунок 16 – «Добавление услуги»

# **ЗАКЛЮЧЕНИЕ**

В результате курсовой работы было проведено исследование предметной области, изучена среда разработки, разработан интерфейс C#, проведено тестирование и отладка программного обеспечения, произведен сбор данных об объектах.

Было составлено техническое задание.

Выбрано и обосновано средство реализации базы данных и программного продукта.

Была создана база данных, которая содержала четыре таблицы.

Был разработан программный продукт.

Разработанный проект позволит всем желающим ознакомится с работой фитнесс-клуба.

В дальнейшем планируется провести более детальную проработку интерфейса приложения.

Цель достигнута, задачи выполнены.

# **СПИСОК ЛИТЕРАТУРЫ**

1. Советов,Б.Я.. Базы данных: Учебник для СПО / Б.Я. Советов, В.В.Цехановский, В.Д. Чертовской. – 2-е изд. –М.: Издательство Юрайт, 2017 – 463 с.
2. Рыбальченко,М.В. Архитектура информационных систем: учеб. пособие для СПО / М.В. Рыбальченко. – М.: Издательство Юрайт, 2017. -91 с.
3. Куприянов Д. В. Информационное обеспечение профессиональной деятельности [Текст]: учебник и практикум для СПО / Д.В. Куприянов. – Москва.: Юрайт, 2017. – 255 с.-(Профессиональное образование). (Гриф).
4. Рудаков А.В. Технология разработки программных продуктов [Текст] :учебник для СПО / А.В. Рудаков.-12- е изд., стер. -Москва :Академия, 2018.-208с.-(Профессиональное образование). (Гриф).
5. Советов Б.Я. Базы данных [Текст]: учебник для СПО / Б. Я. Советов, В.В. Цехановский, В.Д.Чертовский .- 2 изд.- Москва : Юрайт, 2017. – 463 с.-(Профессиональное образование). (Гриф).

**ПРИЛОЖЕНИЯ**