

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Тольяттинский государственный университет»

Институт математики, физики и информационных технологий
(наименование института полностью)

Кафедра «Прикладная математика и информатика»
(наименование)

09.03.03 Прикладная информатика
(код и наименование направления подготовки, специальности)

Бизнес-информатика
(направленность (профиль) / специализация)

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)

на тему «Разработка информационной системы автоматизации деятельности администратора физкультурно-оздоровительного центра»

Обучающийся _____ Д.А. Приданников _____
(И.О. Фамилия) (личная подпись)

Руководитель _____ Е.А. Ерофеева _____
(ученая степень, звание, И.О. Фамилия)

Тольятти 2022

Аннотация

Выпускная квалификационная работа состоит из 51 страницы, 24 рисунка, 3 таблиц, 28 источников литературы.

Темой выпускной квалификационной работы является разработка информационной системы автоматизации деятельности администратора физкультурно-оздоровительного центра, предназначенная для повышения эффективности работы администратора, который занимается регистрацией клиентов и распределением на занятия.

Автоматизация деятельности администратора является актуальной в связи с тем, что на текущий момент регистрация клиентов не осуществляется и данные о нем нигде не хранятся, запись на занятия происходит в журнале, где ведется запись в группы на определенные физкультурно-оздоровительные занятия.

Цель работы – разработка информационной системы автоматизации деятельности администратора физкультурно-оздоровительного центра.

Задачи работы:

- провести изучение деятельности физкультурно-оздоровительного центра и выявить узкие места в бизнес-процессах;
- провести проектирование взаимодействия информационной системы, разработать структуру базы данных;
- реализовать информационную систему физкультурно-оздоровительного центра, написать руководство пользователя.

В первой главе проведено исследование деятельности администратора физкультурно-оздоровительного центра. Рассмотрены основные функции и задачи на рабочем месте в течении рабочего дня. Выполнено моделирование предметной области с помощью методологии IDEF0.

На втором этапе выполнения работы осуществлено описание информационной системы, входящих и исходящих информационных потоков.

Рассмотрены нормативно-справочные документы. Описана структура базы данных и инструменты разработки.

На третьем этапе завершающем описан процесс разработки информационной системы, представлен интерфейс системы для администратора физкультурно-оздоровительного центра.

При выполнении работы были использованы различные инструменты для моделирования и проектирования.

Проектирование организационной структуры было выполнено с помощью программного продукта ARIS Express. Моделирование бизнес-процессов выполнялось с помощью инструментов программы Ramus Education.

Проектирование диаграмм, основанное на объектно-ориентированном подходе, было выполнено с помощью программного продукта StarUML.

Схемы моделей данных были смоделированы с помощью программного продукта AllFusion ERwin Data Modeler.

Содержание

Введение.....	5
1 Исследование предметной области	7
1.1 Анализ деятельности физкультурно-оздоровительного центра.....	7
1.2 Выделение бизнес-процессов организации	9
1.3 Постановка задачи.....	15
1.4 Анализ существующих разработок	17
2 Проектирование информационной системы	22
2.1 Выбор технического обеспечения и средств разработки.....	22
2.2 Характеристика входной и выходной информации.....	25
2.3 Описание структуры базы данных	26
3 Разработка информационной системы для администратора физкультурно-оздоровительного центра	33
3.1 Функции системы и описание интерфейса информационной системы	33
3.2 Структурная схема разрабатываемой системы	36
3.3 Описание программных модулей	39
3.4 Схема работы информационной системы.....	41
3.5 Тестирование информационной системы.....	43
3.6 Руководство пользователя.....	44
Заключение	47
Список используемой литературы	48

Введение

Выпускная квалификационная работа посвящена разработке информационной системы деятельности администратора физкультурно-оздоровительного центра.

Каждый человек старается поддерживать форму. С самого детства сначала родители приводят своих детей на различные занятия спортом, затем уже в сознательном возрасте люди сами выбирают чем они хотят заниматься в свободное время. В стране для этого предназначены много физкультурно-оздоровительных центров, как бюджетных, так и на коммерческой основе [1].

Оздоровление организма происходит разными способами, наиболее естественный занятия спортом. Множество клубов и центров предлагают большой ассортимент услуг для женщин, мужчин и детей. Таким образом поток клиентов постоянно увеличивается, также на это может влиять и сезонность. Администратор должен обеспечить бесперебойную работу тренеров, правильно распределяя посетителей центра на занятия.

Чтобы повысить трафик посещаемости физкультурно-оздоровительного центра необходимо соответствовать новым тенденциям в сфере обслуживания и применять все методы для того, чтобы клиент чувствовал себя, достаточно комфортно приходя заниматься спортом в центр и не тратил попусту время на ожидание, к примеру оформления абонеента.

Цель работы – разработка информационной системы автоматизации деятельности администратора физкультурно-оздоровительного центра.

Задачи работы:

- провести изучение деятельности физкультурно-оздоровительного центра и выявить узкие места в бизнес-процессах;
- провести проектирование взаимодействия информационной системы, разработать структуру базы данных;
- реализовать информационную систему физкультурно-оздоровительного центра, написать руководство пользователя.

Объектом исследования является физкультурно-оздоровительный центр.

Предметом исследования является разработка информационной системы администратора физкультурно-оздоровительного центра.

В первой главе будет проведено исследование деятельности администратора физкультурно-оздоровительного центра. Рассмотрены основные функции и задачи на рабочем месте в течении рабочего дня. Выполнено моделирование предметной области с помощью методологии IDEF0.

На втором этапе выполнения работы будет осуществляться описание информационной системы. Описаны входящая и исходящая информационные потоки. Рассмотрены нормативно-справочные документы. Описана структура базы данных и инструменты разработки.

На третьем этапе описан процесс разработки информационной системы, представлен интерфейс системы для администратора физкультурно-оздоровительного центра.

Разработанную информационную систему можно будет внедрить в деятельность небольшого центра, а также обновлять функционал системы по мере необходимости при расширении физкультурно-оздоровительного центра.

1 Исследование предметной области

1.1 Анализ деятельности физкультурно-оздоровительного центра

Предметной областью настоящей выпускной квалификационной работы является деятельность физкультурно-оздоровительного комплекса (ФОК).

Целью деятельности ФОКа является повышение качества проведения и реализации городских программ физкультурно-массовой направленности, осуществление социально-экономических, спортивно-оздоровительных проектов, ориентированных на массовое оздоровление детей, подростков и взрослого населения занятий, учащихся спортивных и общеобразовательных школ.

Назначение ФОКа — проведение полноценных учебно-тренировочных занятий, в том числе для маломобильных групп населения, по различным спортивным дисциплинам [2].

Функции ФОКа в рамках физкультурно-оздоровительной деятельности:

- реализация учебно-тренировочного процесса, организация соревнований и сборов, здорового досуга и отдыха, проведение лагерных оздоровительных кампаний;
- осуществление социально-экономических, спортивно-оздоровительных проектов, направленных на массовое оздоровление детей, подростков, учащихся и взрослого населения;
- планирование спортивных и спортивно-массовых мероприятий, лагерных оздоровительных кампаний;
- планирование и осуществление воспитательной работы с занимающимися;
- контроль учебно-тренировочного процесса, контроль исполнения учебных планов и программ;
- обеспечение взаимодействия с физкультурно-спортивными вузами, средними специальными учебными заведениями, физкультурно-

спортивными организациями, спортивными клубами в части физкультурно-оздоровительной и спортивной работы.

На рисунке 1 показана организационная структура физкультурно-оздоровительного центра.



Рисунок 1 – Организационная структура физкультурно-оздоровительного центра

Функции администратора центра заключаются в следующем [3]:

- вести контроль пропускной системы;
- оформление и продажа абонементов клиенту;
- ввод данных клиента и регистрация при необходимости;
- консультация по занятиям;
- формирование групп на занятия [4].

Комплекс постоянно расширяет спектр оказываемых услуг и заключает договора только с проверенными тренерами, которые имеют соответствующую репутацию на рынке.

1.2 Выделение бизнес-процессов организации

Рассмотрим деятельность администратора в течении рабочего дня, выполняемые им бизнес-процессы в течении рабочего дня с помощью методологии IDEF0 [5].

Моделирование будем осуществлять с помощью программного продукта Ramus Education.

Контекстная диаграмма представлена на рисунке 2.

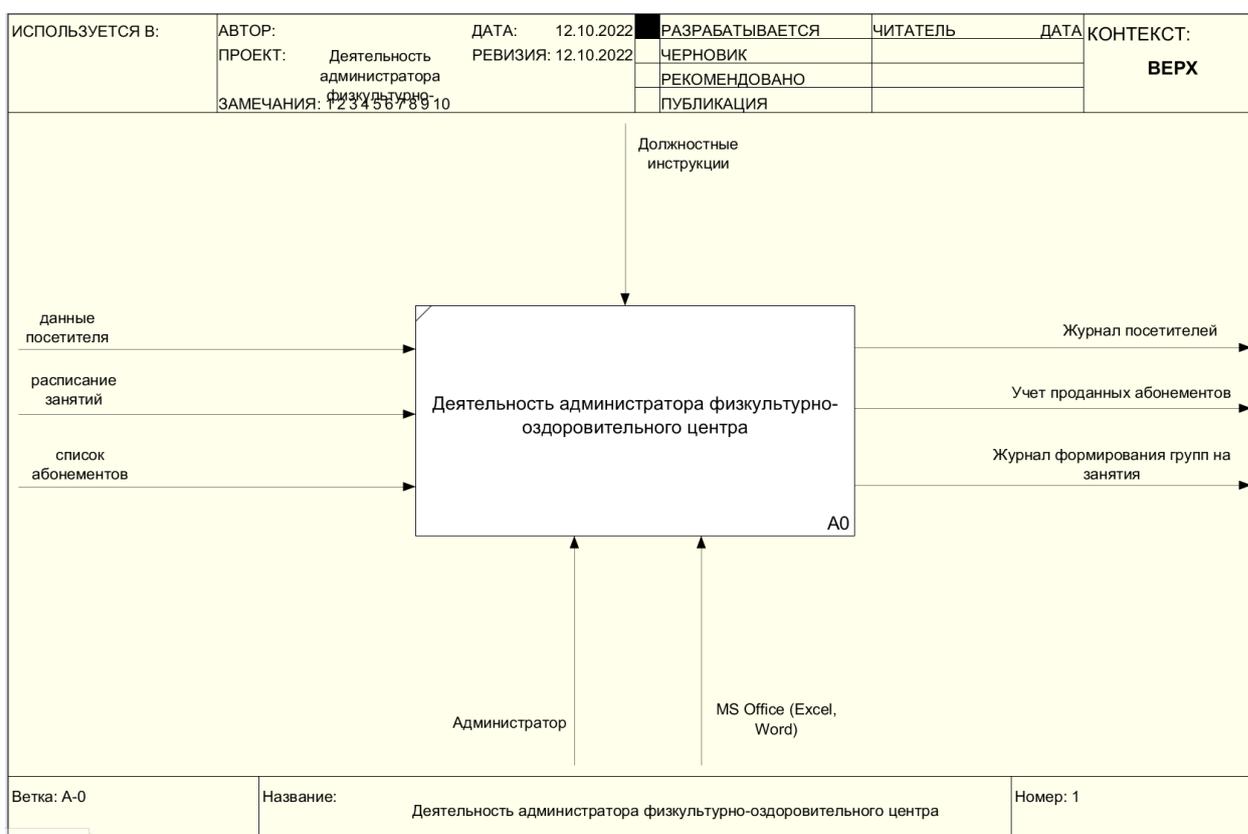


Рисунок 2 – Контекстная диаграмма «как есть» деятельности администратора физкультурно-оздоровительного центра

Входящими потоками, с которыми работает администратор каждый день будут информация о посетителе при регистрации. При обработке данных при информировании посетителя о занятиях администратор основывается на действующее расписание и наличии абонементов.

Выходными потоками после проведения регистрации посетителя, оформления и продажи абонеента, формирования групп на занятия будут журнал посетителей, учет проданных абонеентов, журнал формирования групп на занятия.

Администратор ведет определенные действия на всех процессах диаграммы, фиксация данных на текущий момент осуществляется в электронные документы Word и Excel.

Декомпозиция процесса представлена на рисунке 3.

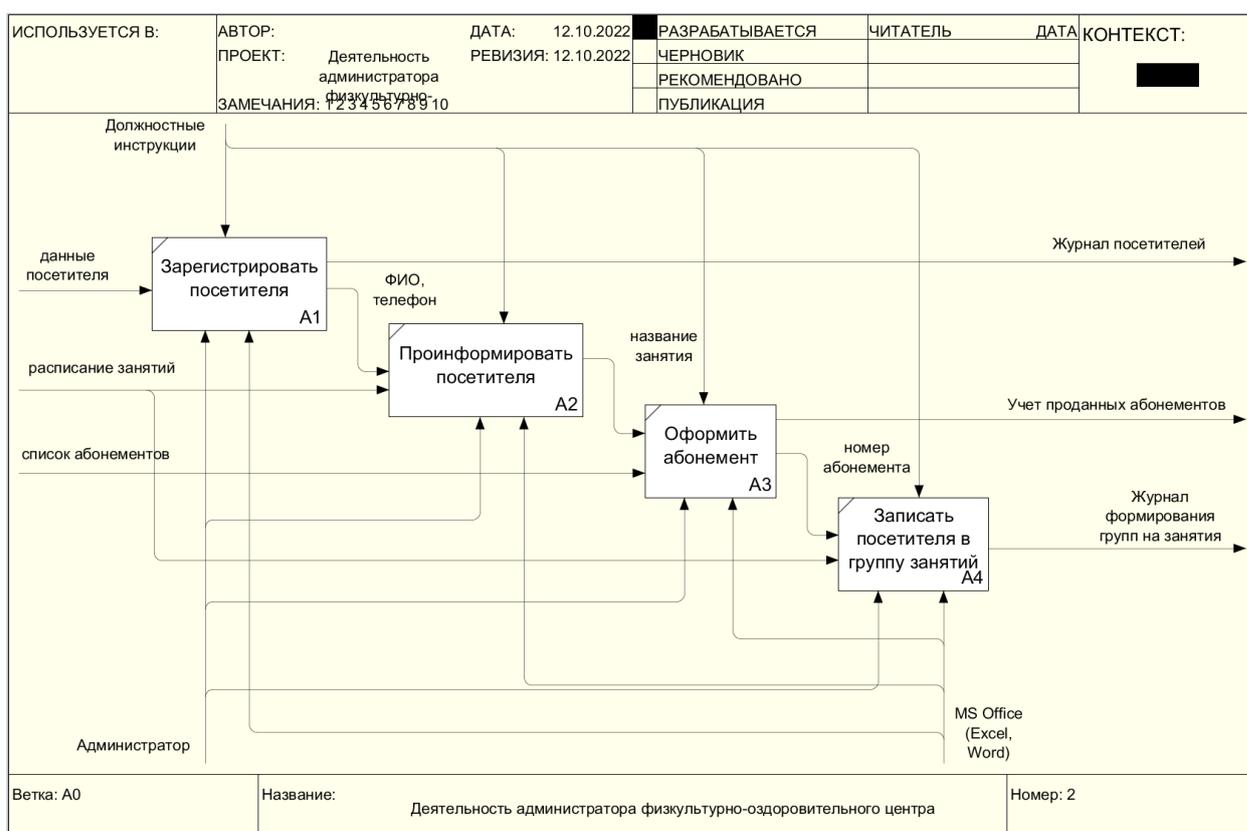


Рисунок 3 – Декомпозиция модели «как есть»

Основные бизнес-процессы, выполняемые администратором физкультурно-оздоровительного центра:

- зарегистрировать посетителя;
- проинформировать посетителя о занятиях;
- оформить и продать абонемент на занятия;

- записать посетителя в группу на занятия.

Посетитель приходит в физкультурно-оздоровительный центр для того, чтобы получить информацию о проводимых занятиях, их расписание и стоимость. Также приобрести абонемент и записаться в группу на занятия.

Диаграмма процесса регистрации посетителя представлено на рисунке 4.

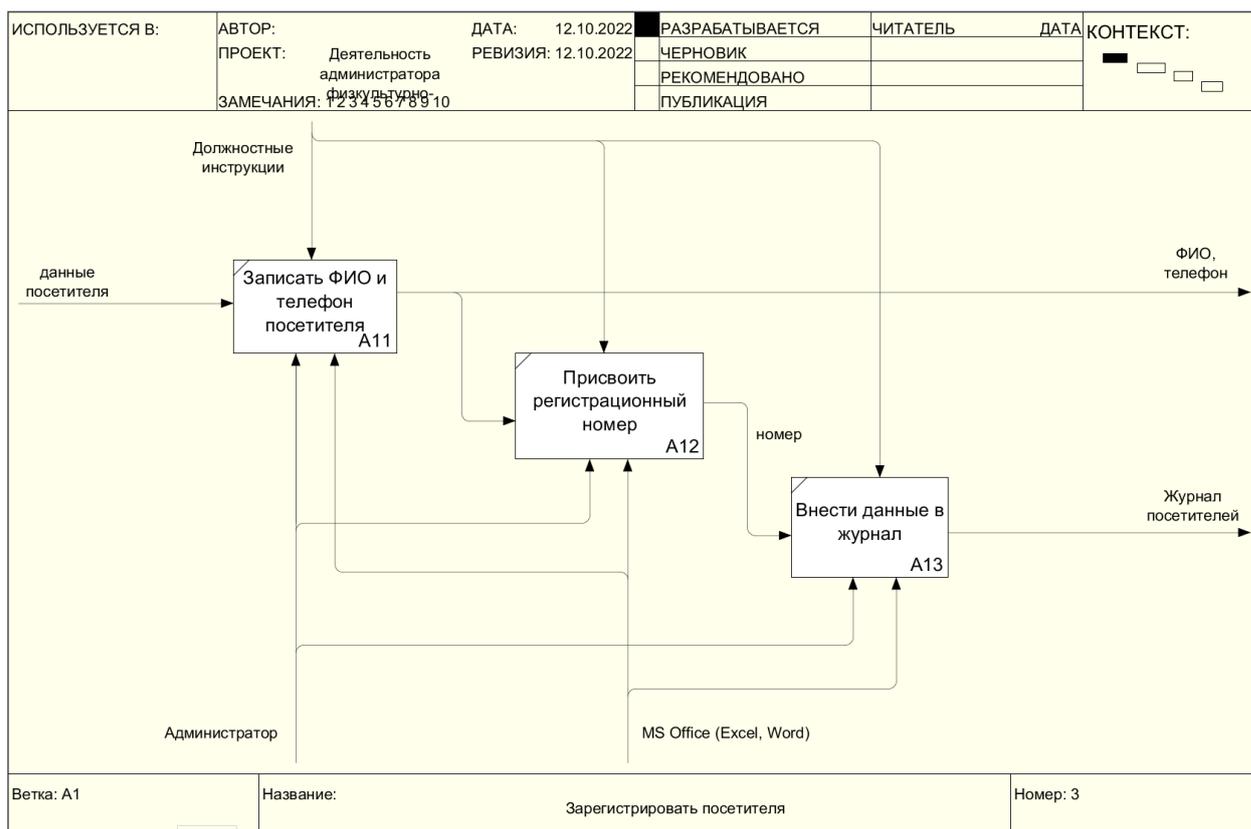


Рисунок 4 – Декомпозиция процесса «Зарегистрировать посетителя»

При посещении физкультурно-оздоровительного центра и выбора занятий его данные регистрируются в журнале для дальнейшего отслеживания по оплате абонемента и продлению при необходимости.

На рисунке 5 представлена декомпозиция блока «Оформить абонемент». Если посетитель определился с видом занятий и готов приобрести абонемент на занятия, администратор должен продать и правильно оформить абонемент, который будет именованным и принадлежать только одному посетителю.

Каждый посетитель может приобрести несколько абонементов на разные занятия.

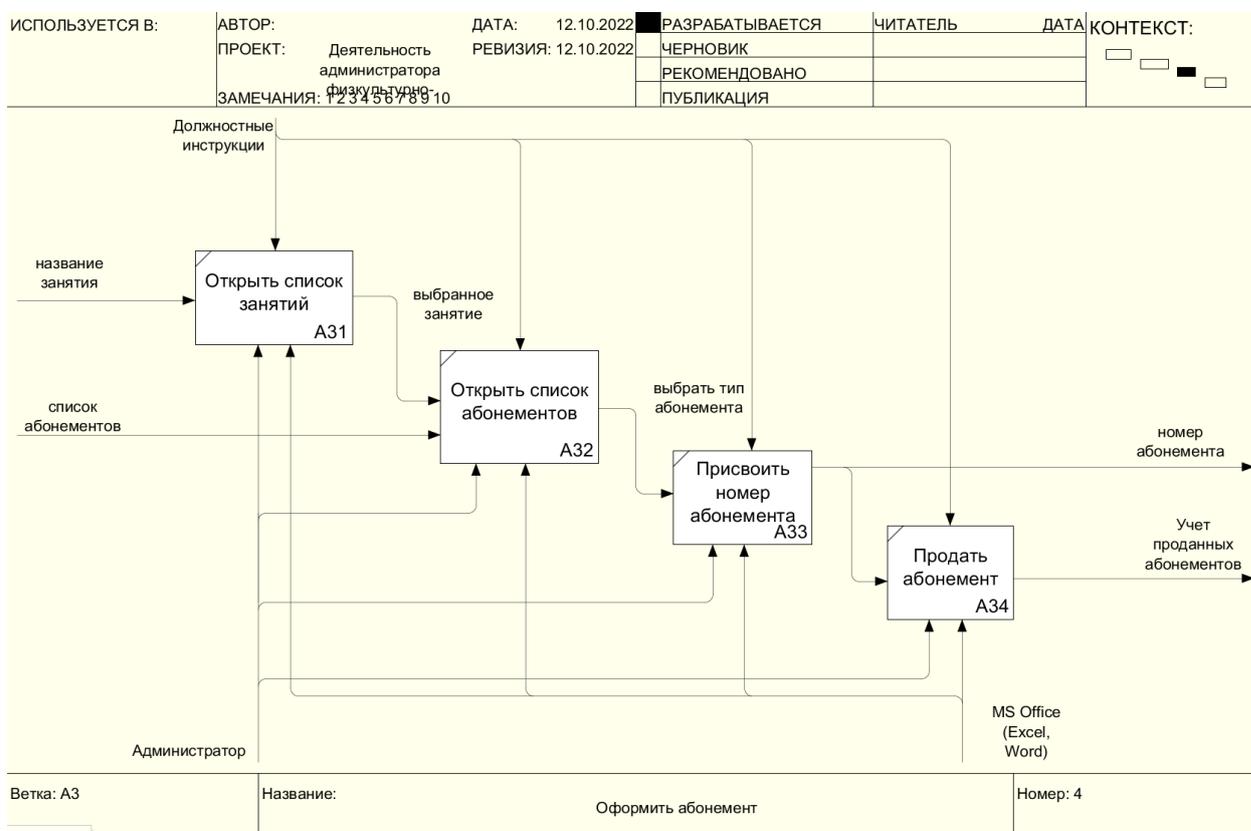


Рисунок 5 – Декомпозиция процесса «Оформить абонемент»

Декомпозиция процесса записи посетителя в группу занятий представлено на рисунке 6.

Если посетитель определился с видом занятий и расписанием их проведения, администратор записывает его в группу, где он будет заниматься с другими клиентами, купившими абонемент на данные занятия.

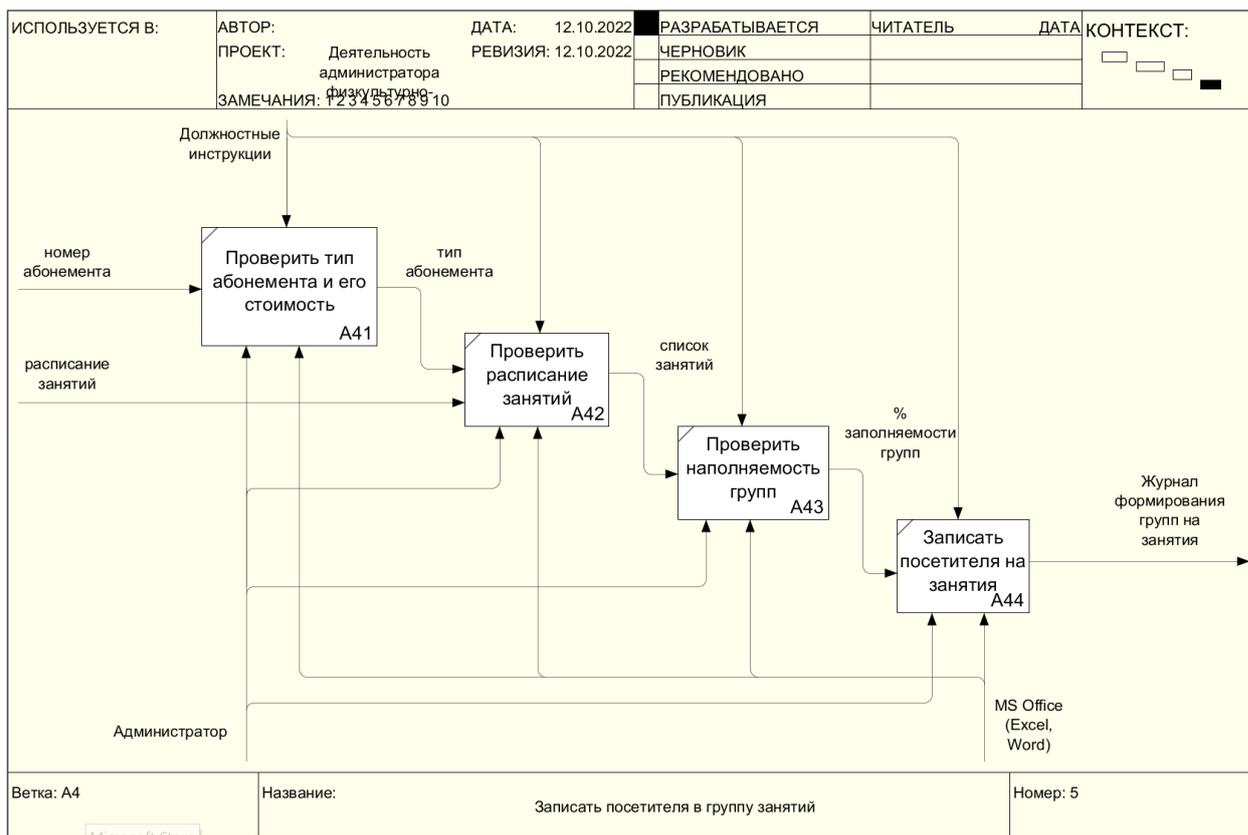


Рисунок 6 – Декомпозиция процесса «Записать посетителя в группу занятий»

Регистрация и оформление абонемента занимает очень много времени у администратора физкультурно-оздоровительного центра, что влечет за собой очереди в сезон при большом потоке посетителей. Разработка и внедрение информационной системы позволит значительно сократить время оформления абонемента и посетителя центра. Эффективность работы администратора будет повышена приблизительно на 45 процентов.

На основе моделирования процессов и декомпозиций процессов можно выделить основные потоки полученной информации и информации, получаемой на выходе при осуществлении определенных функций и процессов.

Входные данные:

- данные посетителя;
- информация об абонементе;

- информация об оплате;
- данные о занятиях;
- данные о расписании;
- данные сформированных групп.

На основе выделенных диаграмм процесса «как есть» можно провести реинжиниринг деятельности администратора путем внедрения информационной системы.

На рисунке 7 показан процесс «как должно быть».

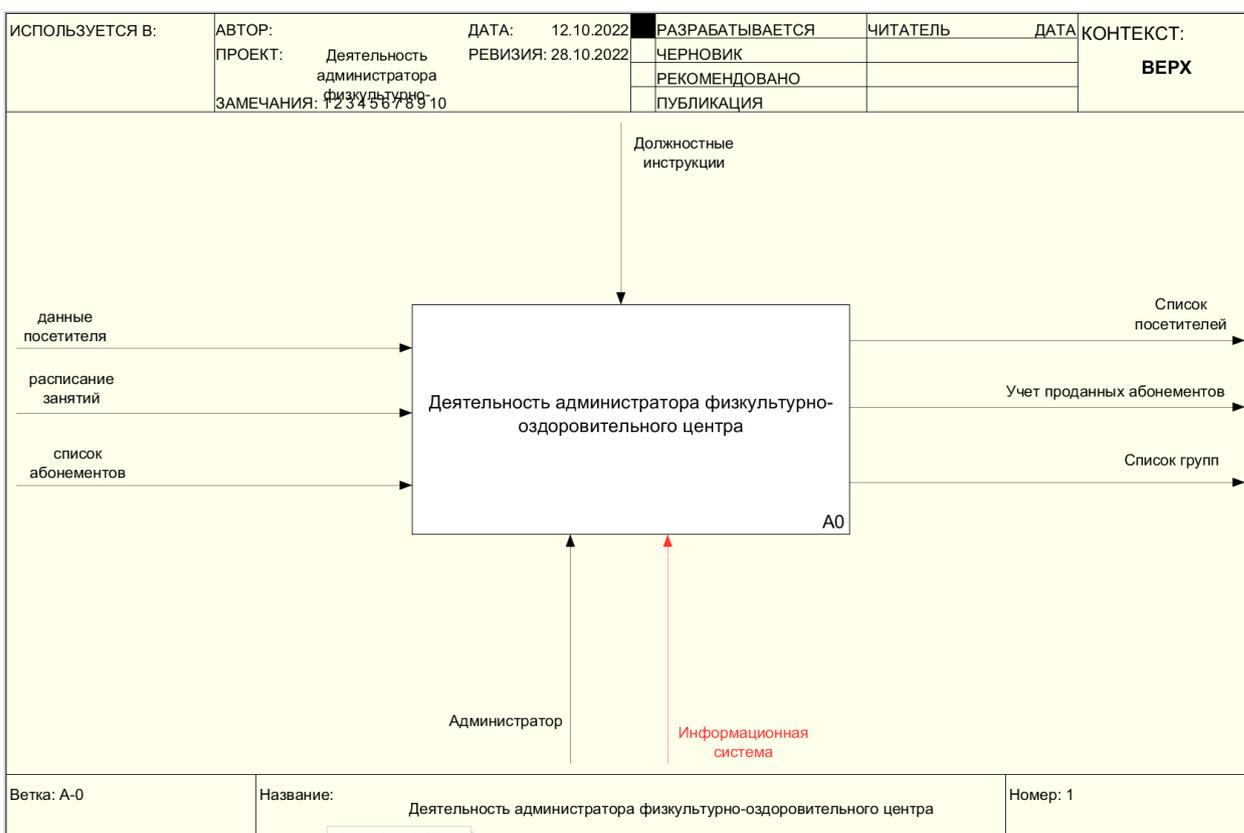


Рисунок 7 – Контекстная диаграмма «как должно быть» деятельности администратора физкультурно-оздоровительного центра

На рисунке 8 показана декомпозиция процесса «как должно быть».

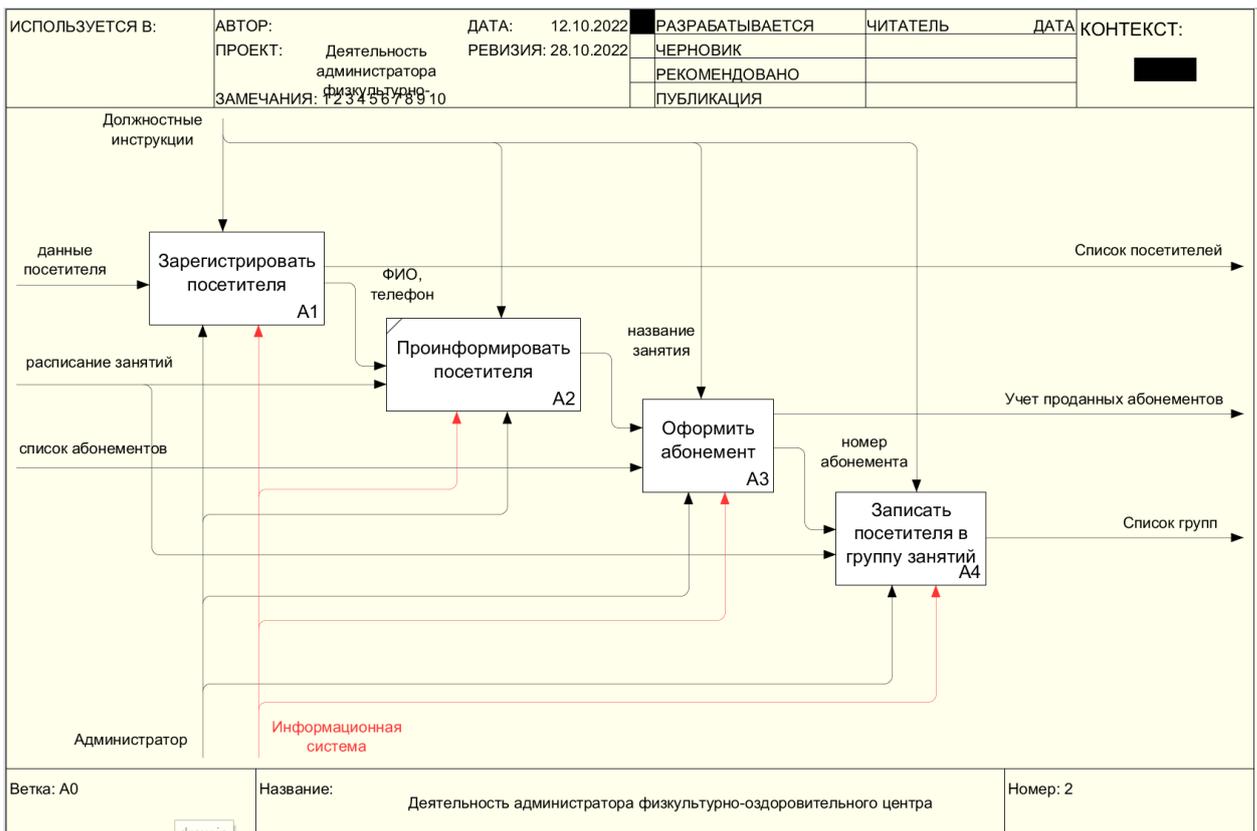


Рисунок 8 – Декомпозиция модели «как должно быть»

На диаграмме показана информационная система с помощью, которой администратор выполняет все бизнес-процессы в течении рабочего дня.

1.3 Постановка задачи

Создание информационной системы предполагает, что основные функции по накоплению, хранению и переработке будут возлагаться на вычислительную технику, а пользователь выполняет часть ручных операций, требующих особый подход к подготовке пользовательских решений. Практически реализуется система «человек-машина».

Для эффективного решения задач администратору необходимо оперативно спланировать свою работу.

Функциональность системы должна обеспечивать выполнение наиболее востребованных операций, таких как:

- возможность вводить данные о посетителях и абонементов в

созданные формы и сохранять их в базу данных;

- возможность формировать группы;
- просмотр и добавление данных в расписание занятий;
- ведение учета по оплате;
- ведение данных о занятиях;
- просмотр списков по заданным критериям.

Требования к АИС:

- надежность и простота обслуживания;
- графический интерфейс;
- наличие базы данных для хранения информации;
- ежедневное резервное копирование данных на сторонний

носитель.

Постоянным пользователем информационной системы будет администратор физкультурно-оздоровительного центра. Вести администрирование системы будет администратор системы, который фактически будет заниматься поддержкой работоспособности системы.

На рисунке 9 представлена диаграмма вариантов использования [6].

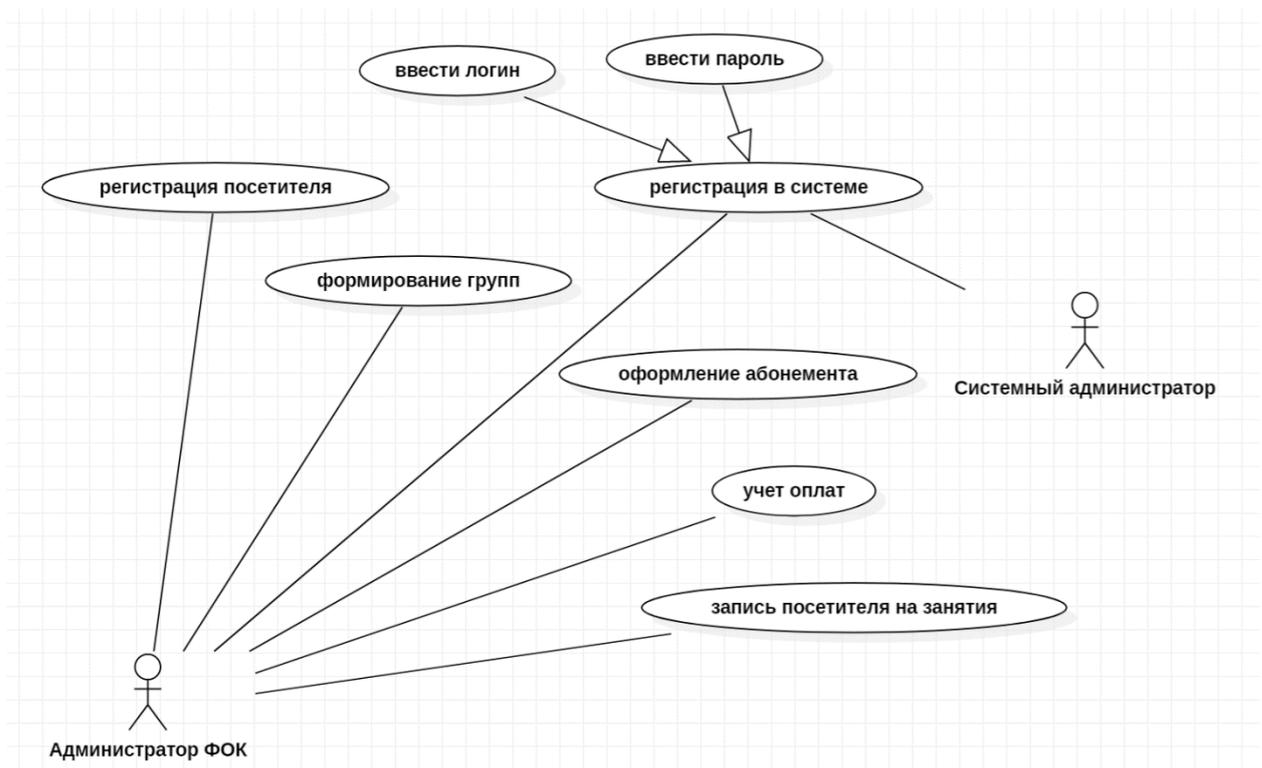


Рисунок 9 – Диаграмма вариантов использования

Основные функции информационной системы:

- учет оплат при оформлении абонемента;
- оформление абонемента;
- регистрация посетителей;
- формирование групп на занятия;
- запись посетителя в группу занятий.

Информационной системой будут пользоваться по мере необходимости в течении дня.

Пользователи информационной системы:

- администратор информационной системы;
- администратор физкультурно-оздоровительного комплекса.

1.4 Анализ существующих разработок

Приведем сравнение автоматизаций деятельности администратора физкультурно-оздоровительного комплекса, программные разработки будут рассмотрены только отечественной разработки.

Проанализируем некоторые достоинства и недостатки программ:

Программа Mobifitness обладает всеми инструментами для поддержания работы фитнес-студии. Специальные функции для удобства работы фитнес-тренера, рецепции, отдела продаж, директора и простого взаимодействия с клиентами [7].

Программное решение для управления фитнес-клубом или небольшой студией.

Создана единая платформа, в которой есть все, что нужно:

- учетная система для легкого управления;
- умная онлайн-касса для ускорения работы администратора;
- личный кабинет и интернет-магазин для увеличения продаж.

Преимуществом является интеграция всех элементов системы в одну, что позволяет отслеживать все процессы.

Пример расписания занятий представлен на рисунке 10.

	Пн 15.04	Вт 16.04	Ср 17.04	Чт 18.04	Пт 19.04	Сб 20.04	Вс 21.04
09:00	09:00 – 10:00 Групп. Аштанга-Йога Зал Йоги и пилатеса Чиботаренко Юлия			09:00 – 10:00 Групп. Аштанга-Йога Зал Йоги и пилатеса Чиботаренко Юлия			09:00 Аштанга-Йога Зал Йоги и пилатеса Чиботаренко Юлия
10:00		10:00 – 11:00 Групп. АБС: пресс+спина Зал групповых программ		10:00 – 11:00 Групп. АБС: пресс+спина Зал групповых программ		10:00 – 11:00 Групп. АБС: пресс+спина Зал групповых программ	
11:00			11:00 – 12:00 Групп. Пилатес 50+ Зал Йоги и пилатеса	11:00 – 12:00 Групп. Пилатес 50+ Зал не указан	11:00 – 12:00 Групп. Пилатес 50+ Зал не указан	11:00 – 12:00 Групп. Пилатес 50+ Зал не указан	11:00 Пилатес 50+ Зал не указан
12:00	12:00 – 13:00 Групп. LOW BODY: ноги и ягодицы Зал групповых программ Сенин Олег		12:00 – 13:00 Групп. LOW BODY: ноги и ягодицы Зал не указан	12:00 – 13:00 Групп. LOW BODY: ноги и ягодицы Зал не указан		12:00 – 13:00 Групп. LOW BODY: ноги и ягодицы Зал не указан	12:00 LOW BODY: ноги и ягодицы Зал не указан
13:00							
14:00							
15:00	15:00 – 16:00 Групп. Pole dance труппы Зал Pole dance Чиботаренко Юлия	15:00 – 16:00 Групп. Pole dance труппы Зал Pole dance Чиботаренко Юлия		15:00 – 16:00 Групп. Pole dance труппы Зал Pole dance Чиботаренко Юлия	15:00 – 16:00 Парс. Тренировка с мастер-тренером Перерыв - 5 минут Тренировочный зал Чиботаренко Юлия Клиент: Сидоров Николай Петрович	15:00 – 16:00 Групп. Pole dance труппы Зал Pole dance Чиботаренко Юлия	15:00 Pole dance труппы Зал Pole dance Чиботаренко Юлия
16:00							

Рисунок 10 – Пример расписания занятий системы Mobifitness

Sport Priority [8]. Идеально подойдет для небольших фитнес-клубов, школ танцев и студий йоги. Для тех, кто не хочет разбираться со сложными системами, а хочет больше времени уделять любимому делу.

Комплексное IT-сопровождение небольших клубов, студий и школ в сфере фитнеса и спорта.

Возможности системы:

- мобильное приложение для смартфона;
- интуитивно понятный интерфейс;
- помощь во внедрении сервиса;
- учет посещений клиентов без бумажных абонементов;
- продажа абонементов прямо в расписании;
- аналитические отчеты, расчет зарплат тренеров;
- отдельные личные кабинеты администратора, тренера и управляющего клубом;
- чаты с клиентами;
- рассылки и push-уведомления;
- генератор электронных подарочных сертификатов.

На рисунке 11 показан интерфейс системы Sport Priority.

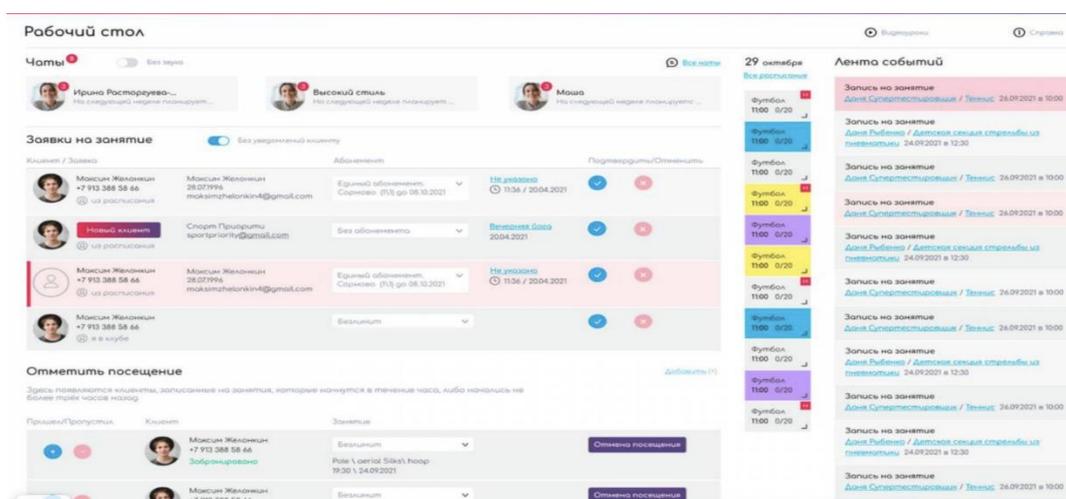


Рисунок 11 – Пример интерфейса системы Sport Priority

FitBase подходит школам танцев, студиям йоги, растяжки, пилатеса, школам единоборств, фитнес-клубам, бассейнам и студиям персональных тренировок [9].

FitBase — программа для управления фитнес-клубом и спортивной студией:

- автоматизация работы;
- CRM с воронками и настройкой автоматических задач;
- виджет расписания и онлайн-запись;
- доступ для всех сотрудников, включая тренеров;
- работает на любых устройствах;
- интегрируется со всем оборудованием в фитнес-клубах;
- мобильное приложение для клиентов;
- мобильное приложение для тренеров;

На рисунке 12 представлен интерфейс системы FitBase.

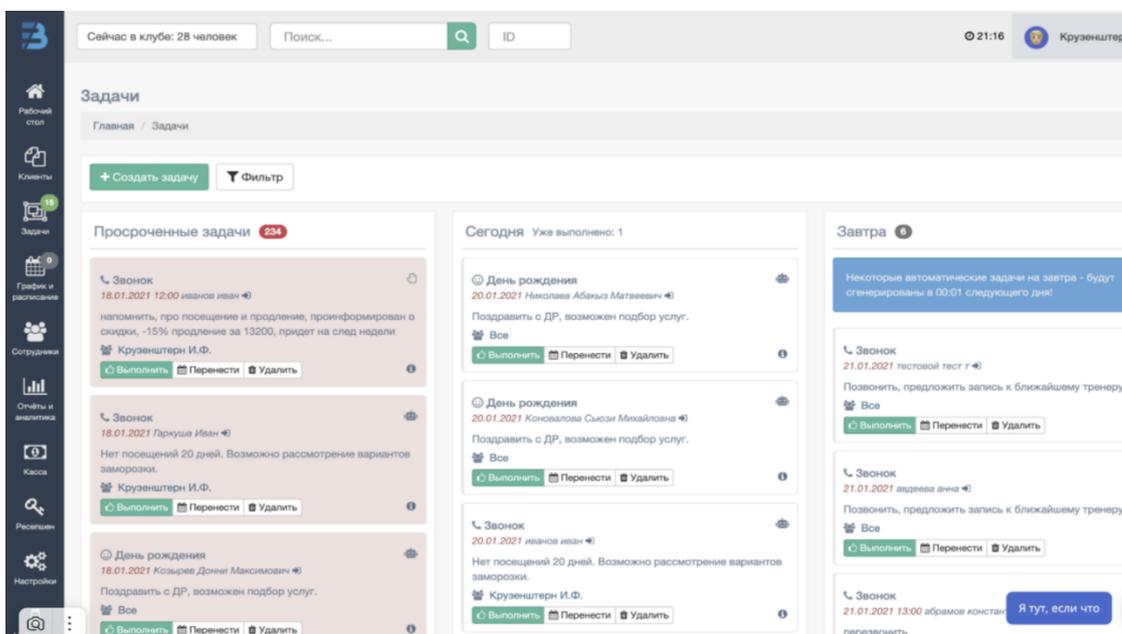


Рисунок 12 – Пример интерфейса системы FitBase

Итоги сравнения программных продуктов показаны в таблице 1.

Таблица 1 – Сравнительная таблица функциональности систем для физкультурно-оздоровительных центров

Показатель сравнения	Mobifitness	Sport Priority	FitBase
1	2	3	4
Простота интерфейса	-	+	+
Возможность ведения учета продаж абонементов	+	+	+
Возможность ведения учета посетителей	+	+	+
Доступность	-	-	-
Защита личных данных	-	-	-
Возможность расширения модуля	+	+	+
Наличие мобильной версии	+	+	-

Таким образом руководство центра приняло решение о создании с нуля программного продукта, который обеспечит выполнение тех функций который будут указаны заказчиком.

Выводы по первому разделу

В первом разделе работы проведено исследование деятельности администратора физкультурно-оздоровительного центра. На основе исследований проведено моделирование бизнес-процессов, что позволило выявить слабые места и предложить методы их автоматизации с помощью внедрения информационной системы. Выполнена постановка задачи на разработку информационной системы. Чтобы разработка системы была целесообразной провели обзор аналогичных систем, предлагаемых на российском рынке, и выяснили, что наилучшим вариантом будет разработать систему с нуля под конкретные требования.

2 Проектирование информационной системы

2.1 Выбор технического обеспечения и средств разработки

Для разработки системы потребуется база данных для хранения информации. Рассмотрим некоторые наиболее распространенные СУБД и выберем подходящую для нашей разработки.

Для сравнения возьмем такие как: MS SQL, Access и PostgreSQL, не будем рассматривать такие как AxisBase, Kexi и подобные [10].

Преимущества СУБД Microsoft Access:

- простой и понятный интерфейс;
- СУБД MS Access общеизвестна и очень популярна в настольных ПК;
- практически безграничные возможности экспорта данных;
- невысокая цена;
- широкий выбор конструкторов для построения форм, отчетов и запросов, с помощью которых можно отфильтровать данные и вывести их в удобном виде;
- возможность создания пароля на своей базе данных;
- встроенный язык VBA высокого уровня.

PostgreSQL (произносится «Пост-Грэс-Кью-Эль») – свободная объектно-реляционная система управления базами данных (СУБД) [11].

PostgreSQL предоставляет множество различных возможностей, достаточно надежна и имеет хорошие характеристики по производительности. Она работает практически на всех UNIX платформах, включая UNIX подобные системы, такие как FreeBSD и Linux. Ее можно применять на Windows NT Server и Windows 2000 Server, а для разработки годятся даже такие системы Microsoft для рабочих станций, как ME. Кроме того, PostgreSQL свободно распространяется и имеет открытый исходный код.

PostgreSQL выгодно отличается от многих других СУБД. Она обладает практически всеми возможностями, которые есть в других базах данных (коммерческих или Open Source), а также некоторыми дополнительными, имеет множество преимуществ и функциональности.

Благодаря многочисленному сообществу пользователей и открытому доступу к исходному коду ошибки исправляются очень быстро, но нам такая СУБД не подходит по ряду причин: отсутствие полной поддержки самых массовых языков программирования VB и C#, потеря форматирования программного кода, также в PostgreSQL модификации объекта, от которого что-то зависит - сделать нельзя вообще. Изменения таблиц, типов возвращаемых параметров функций возможно только после предварительного удаления всей пирамиды объектов, зависящих от изменяемого объекта. С последующей загрузкой всей пирамиды объектов заново, а это очень трудоемкий процесс.

MS SQL является надежной базой данных для любых целей, может продолжать расширяться по мере наполнения информацией, без заметного уменьшения быстродействия операций с записями в многопользовательском режиме. Пользователи могут быть добавлены путем модернизации оборудования. В последнем тесте поддерживалось до 4600 пользователей базы данных.

MS SQL обрабатывает запросы от пользователей и только отправляет пользователю результаты запроса.

Техническое обслуживание MS SQL очень простое и не требует больших знаний [12].

На основе вышеизложенных данных о каждой СУБД сделаны выводы о том, что такие СУБД как Access и PostgreSQL не подходят по своим параметрам для написания информационной системы, в силу того что первая не имеет высокого уровня защищенности и данные организации могут попасть к злоумышленникам, а вторая СУБД не имеет, к сожалению, стандартного адаптера для подключения. Так как задача стоит в написании

информационной системы с выбранным языком С#, эти СУБД не удовлетворяют требованиям.

Представим СУБД в сравнительной таблице.

Таблица 2 – Результаты сравнения баз данных

Основные отличия	Access	MS SQL	PostgreSQL
1	2	3	4
Доступность	-	+	+
Хранение большого объема данных	-	+	+
Скорость обработки операций с большими данными	-	+	-
Защита от несанкционированного доступа	-	+	+
Простота использования	+	-	-

Для написания системы нам нужно выбрать оптимальный язык программирования, чтобы реализовать весь функционал. В качестве языка программирования будет использован язык С#.

Для реализации приложения выбрана интегрированная среда разработки программного обеспечения Visual Studio 2019.

Диаграмма размещения [13] показана на рисунке 13, где показаны основные узлы системы, взаимодействующие между собой в процессе работы.

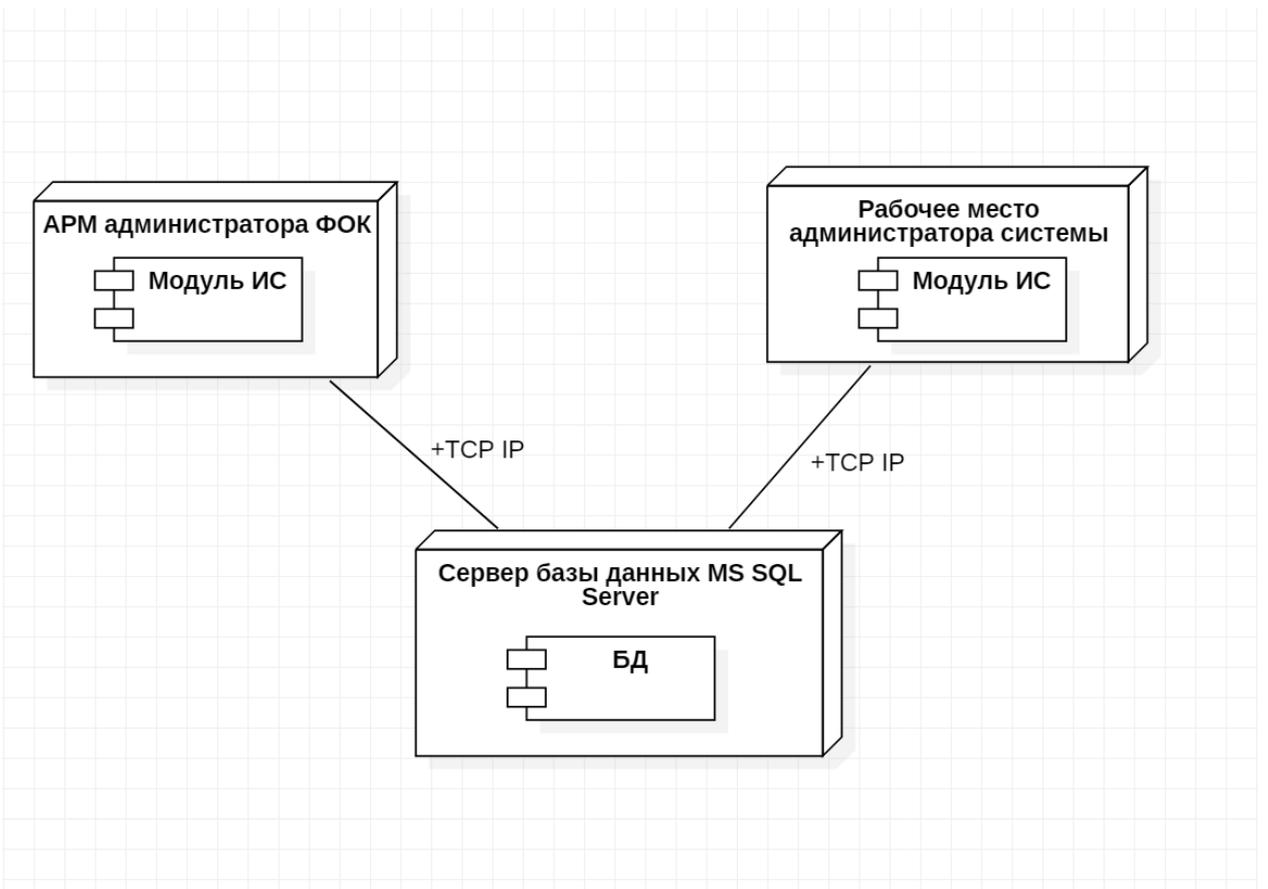


Рисунок 13 – Диаграмма размещения

Технические требования, которые должны быть соблюдены при установке клиенткой части информационной системы [14]:

- оперативная память не менее 4 Гб;
- объем свободного места на жестком диске – 10 Гб.

Минимальные технические требования к серверу, на котором будет установлена СУБД с развернутой базой данных:

- оперативная память не менее 12 Гб;
- объем свободного места на жестком диске – 500 Гб.

2.2 Характеристика входной и выходной информации

В таблице 3 представлены реквизиты входных документов.

Таблица 3 – Характеристика входной информации

Входные данные	Объем в год	Источники получения
Данные о посетителе	1000-1500	Личные данные посетителя
Данные об абонементе	2000-3000	Администрация
Список занятий	30-40	Каталог занятий
Расписание занятий	30-40	Утвержденное расписание
Данные о группах	30-40	Администрация
Данные оплаты	2000-3000	Бухгалтерия

Входящая информация в виде документов представлена в таблице 4.

Таблица 4 – Описание выходных документов

Документ	Форма представления	Назначение
Группы занятий	Бумажный, электронный	Администрация
Отчет об оплате	Бумажный, электронный	Администрация
Продажи абонементов	Бумажный, электронный	Администрация

Исходящие документы администратор может менять в зависимости от получения данных.

2.3 Описание структуры базы данных

Проведение анализа входящей и исходящей информации, получаемой администратором на протяжении рабочего дня, помогли выделить основные информационные объекты, которые описывают процессы деятельности администратора физкультурно-оздоровительного центра.

Для учета посетителей потребуются сведения, которые будут храниться в таблице «Посетитель» и содержать информацию о посетителе: «Код посетителя», «Фамилия», «Имя», «Отчество», «Телефон».

Чтобы посетитель имел достаточно информации о занятиях и абонементы должна храниться информация «Абонемент» и «Занятия».

Информация об абонементе будет содержать данные «Номер абонемена», «Код типа абонемена», «Стоимость».

Чтобы исключить дублирования информации в таблице выделим объект «Тип абонемена» с данными «Код типа абонемена», «Наименование», «Срок действия».

Для осуществления продажи абонементов необходимо хранить информацию «Продажа», которая будет содержать сведения о посетителях и об абонементе, «Код продажи», «Дата продажи», «Номер абонемена», «Количество», «Код посетителя».

Формирование групп происходит на основе данных «Расписание», «Занятия», «Тренерский состав».

Каждая сущность содержит данные: «Расписание» содержит информацию «Номер расписания», «Табельный номер», «Код занятия», «Номер группы».

Сущность «Занятия» содержит данные «Код занятия», «Наименование», «Продолжительность».

Сущность «Тренерский состав» содержит данные «Табельный номер», «Фамилия», «Имя», «Отчество», «Отчество», «Стаж работы», «Разряд».

После анализа данных расписания и занятий, администратор может включить посетителя в ту или иную группу, которая будет содержать информацию «Номер группы», «Код посетителя», «Дата начала занятий», «Количество мест», «Количество свободных мест».

В отношении Абонемент входит в Продажи, добавляются атрибуты: Номер абонемена и Код посетителя, отношение 1:М.

В отношении Тип абонемена принадлежит Абонементу, добавляются атрибуты: Код типа абонемена, отношение 1:М.

В отношении Посетитель входит в Группы, добавляется атрибут: Код посетителя, отношение 1:М.

В отношении Группы входят в Расписание, добавляется атрибут: Номер группы, отношение 1:М.

В отношении Тренерский состав входит в Расписание, добавляется атрибут Табельный номер, отношение 1:М.

В отношении Занятия входит в Расписание, добавляется атрибут Табельный номер, отношение 1:М.

На рисунке 14 показана ER-модель [14].

На рисунке 15 представлена даталогическая модель БД [15]. Физическая модель базы данных для СУБД SQL Server представлена на рисунке 16.

Физическая модель данных – это представление структуры данных, реализованной или предназначенной для реализации в системе управления базами данных. В жизненном цикле проекта она обычно является производной от логической модели данных, хотя может быть переработана из данной реализации базы данных. Полная физическая модель данных будет включать все артефакты базы данных, необходимые для создания связей между таблицами или для достижения целей производительности, такие как индексы, определения ограничений, связывающие таблицы, секционированные таблицы или кластеры. Аналитики обычно могут использовать физическую модель данных для расчета оценок объема хранилища; она может включать конкретные сведения о распределении хранилища для данной системы баз данных.

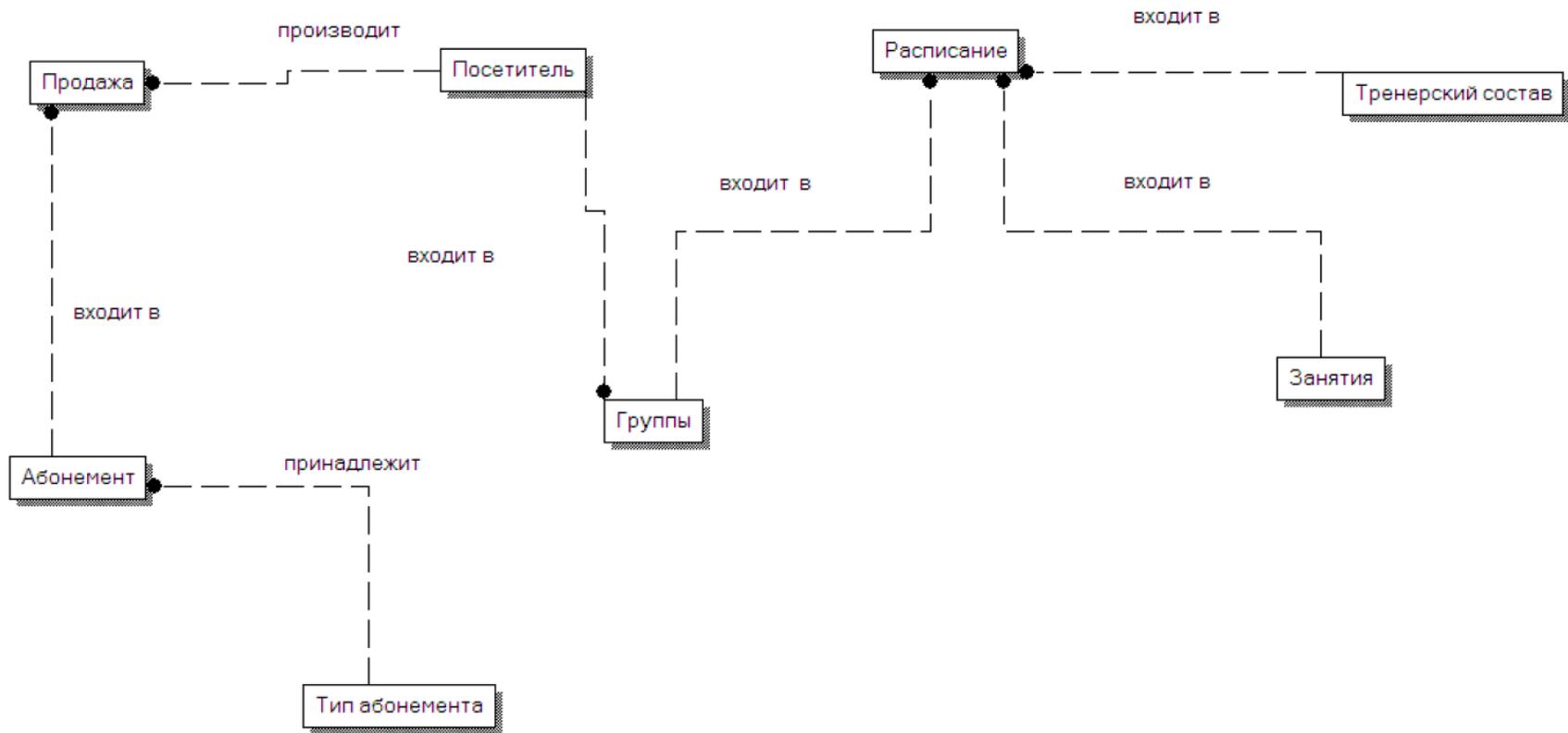


Рисунок 14 – ER-модель данных

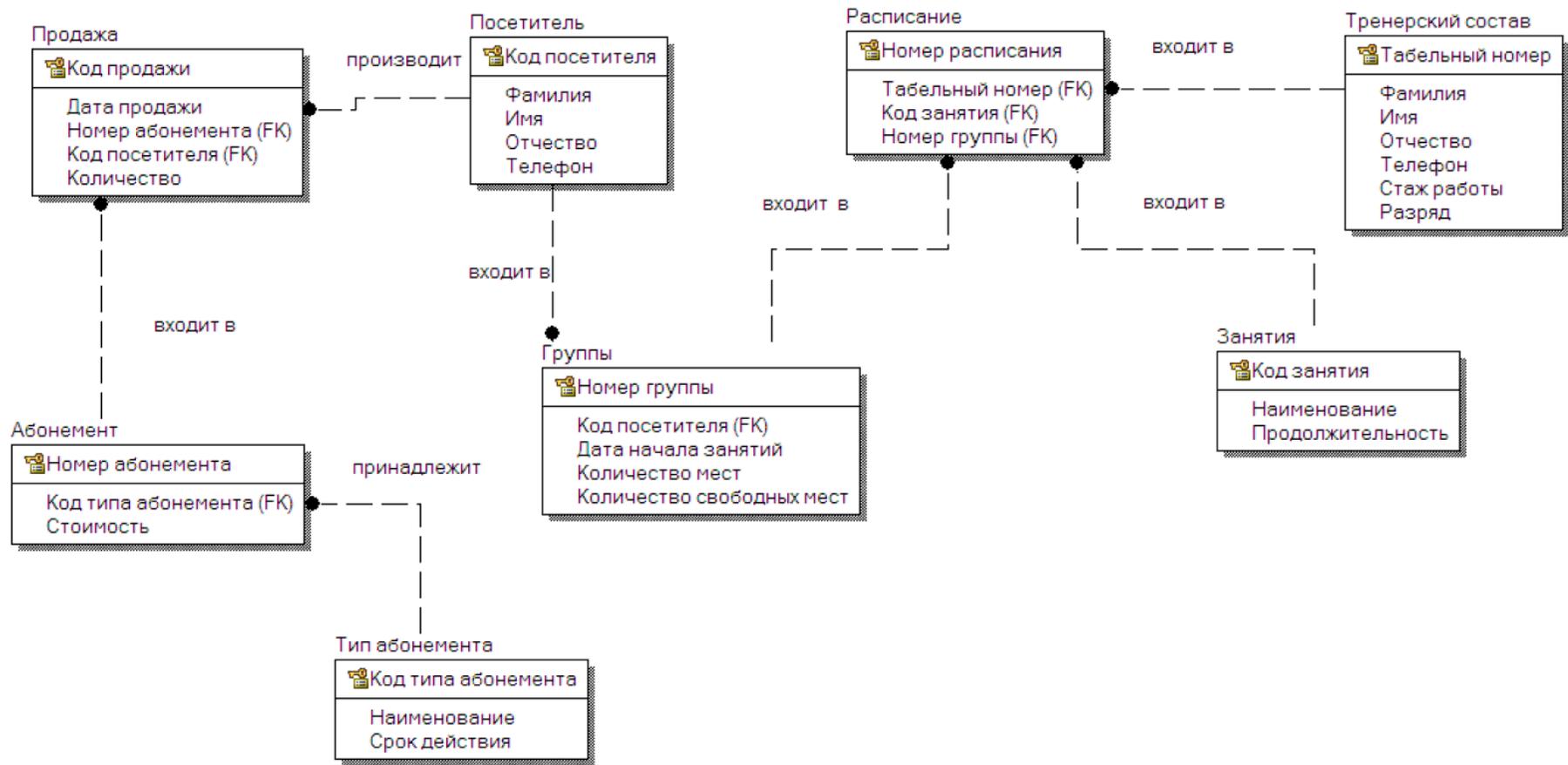


Рисунок 15 - Дatalogическая модель данных

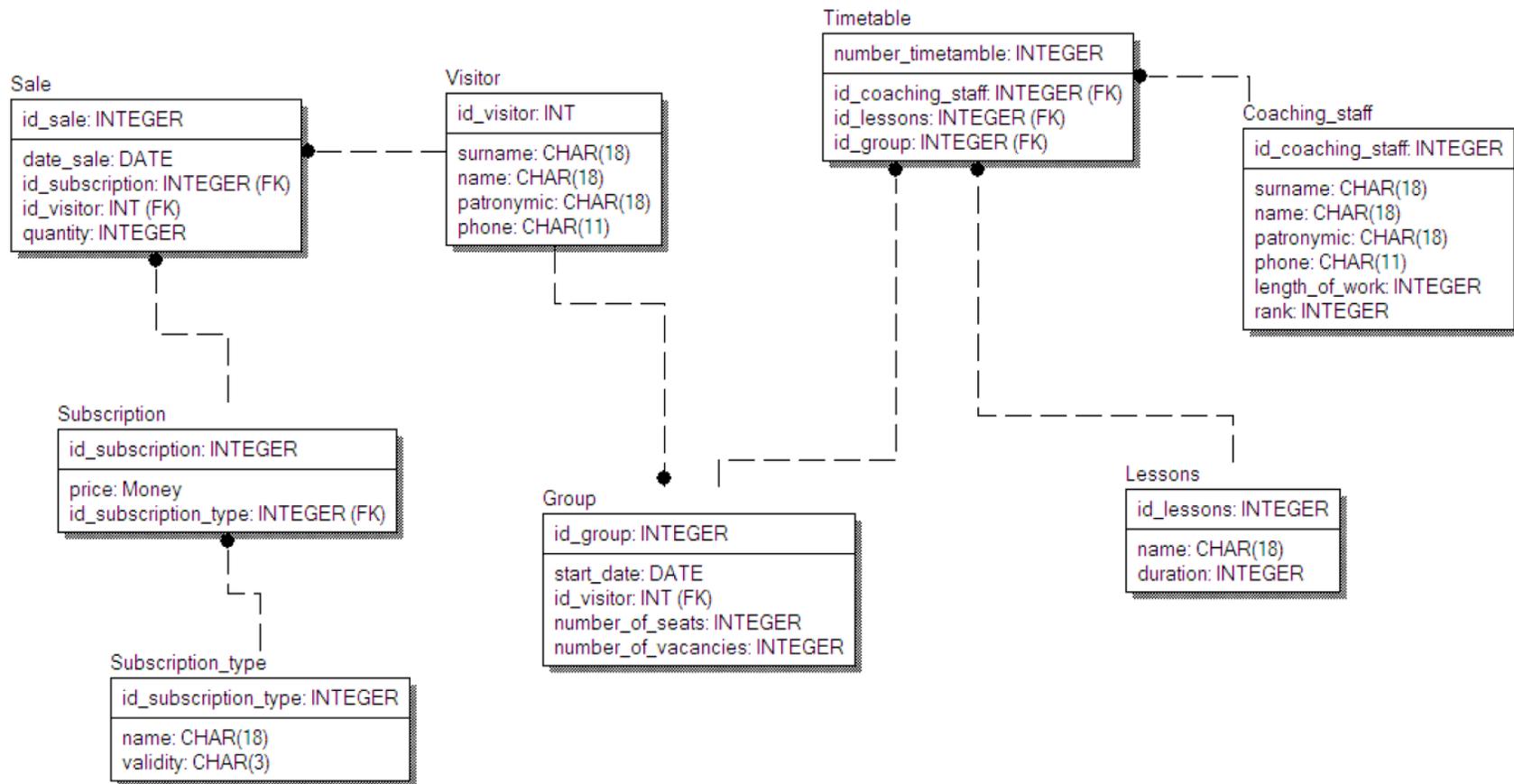


Рисунок 16 - Физическая модель данных

Логическая, даталогическая и физическая модели были разработана с использованием программы AllFusion Data Modeler 7.3.

Приложение было создано для широкой аудитории, поэтому подходит для разработки бизнес-проектов как для мелких компаний, так и для крупных организаций. Среди основных функций инструмента можно выделить следующие:

- работа с текстовыми файлами и их коррекция в автоматическом режиме;
- широкий выбор инструментов, удобно расположенных на панели;
- использование функциональных подсказок;
- разработка схем различных уровней сложности;
- построение графиков и диаграмм;
- использование различных инструментов для создания рисунка;
- разработка планов, сложных моделей и различных структур;
- ведение отчетов и их сохранение в виде электронных файлов.

Выводы по второму разделу

Выполнение второго раздела включает такие работы как проектирование информационной системы, которое необходимо для решения поставленной задачи. В разделе также рассмотрены данные входящей и исходящей информации, с которой работает администратор физкультурно-оздоровительного центра в течении рабочего дня. Проведено моделирование структуры базы данных на уровне логического, даталогического и физического проектирования.

3 Разработка информационной системы для администратора физкультурно-оздоровительного центра

3.1 Функции системы и описание интерфейса информационной системы

Основываясь на разработанные требования к информационной системе проведенное проектирование базы данных и системы в целом, можно выделить функции, которые будут возможны для осуществления деятельности физкультурно-оздоровительного центра:

- ведение данных о посетителях;
- ввод и хранение данных о занятиях;
- ведение информации о расписании занятий;
- ведение данных о тренерском составе;
- ввод данных об абонеентах и их регистрация;
- ведение и формирование данных о группах.

На диаграмме представлены основные функции:

- работа с посетителями;
- работа с абонеентами;
- работа с расписанием;
- работа с группами.

Основные функции имеют дополнительные разветвления на более мелкие работы и функции.

На рисунке 17 показана диаграмма функций информационной системы.

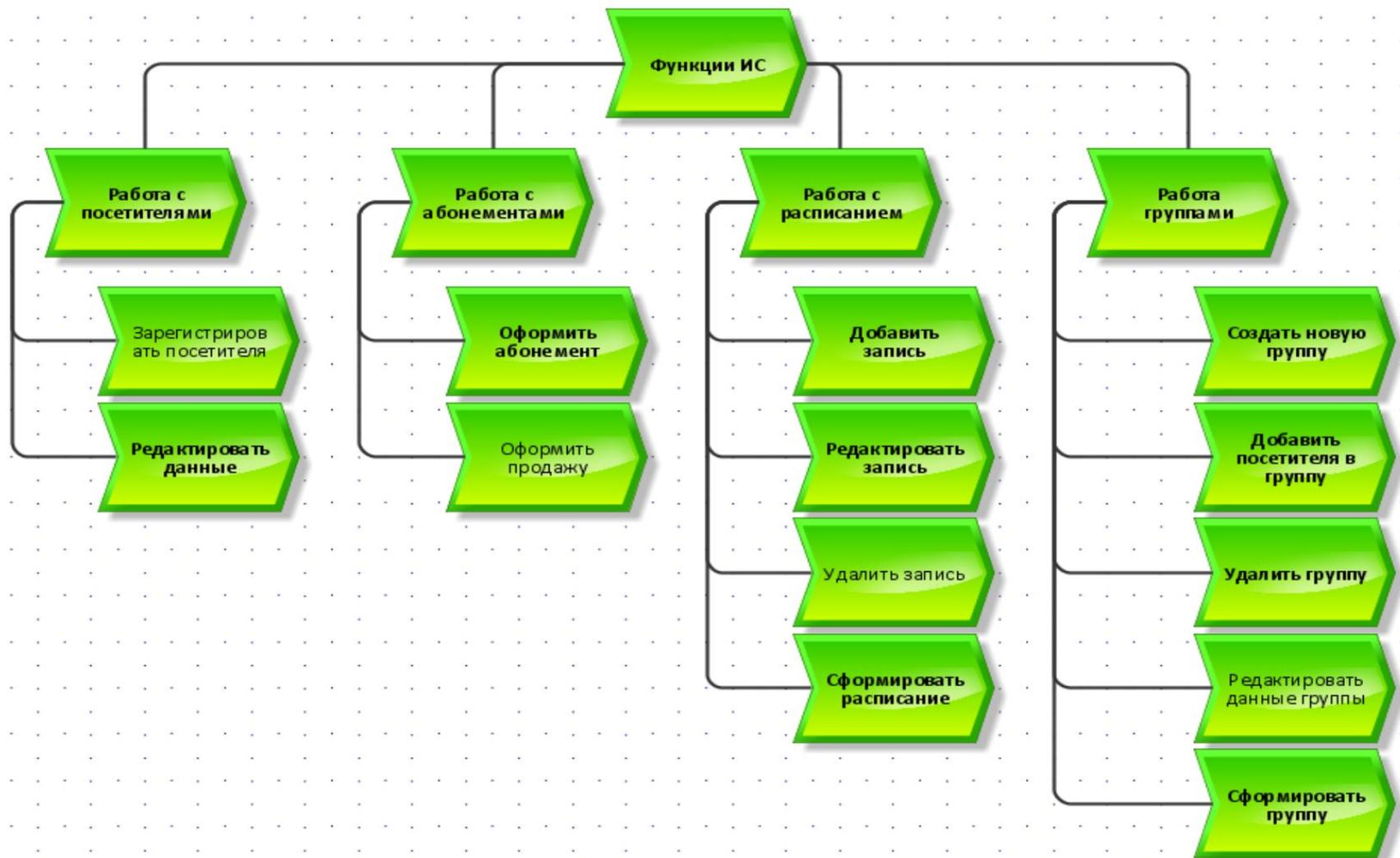


Рисунок 17 – Функциональная диаграмма информационной системы

Для определения общих алгоритмов обработки информации в АИС была построена диаграмма последовательности (Рисунок 18) – взаимодействия языка UML, которые описывают отношения объектов в различных условиях. Условия взаимодействия задаются сценарием, полученным на этапе разработки диаграмм вариантов использования.

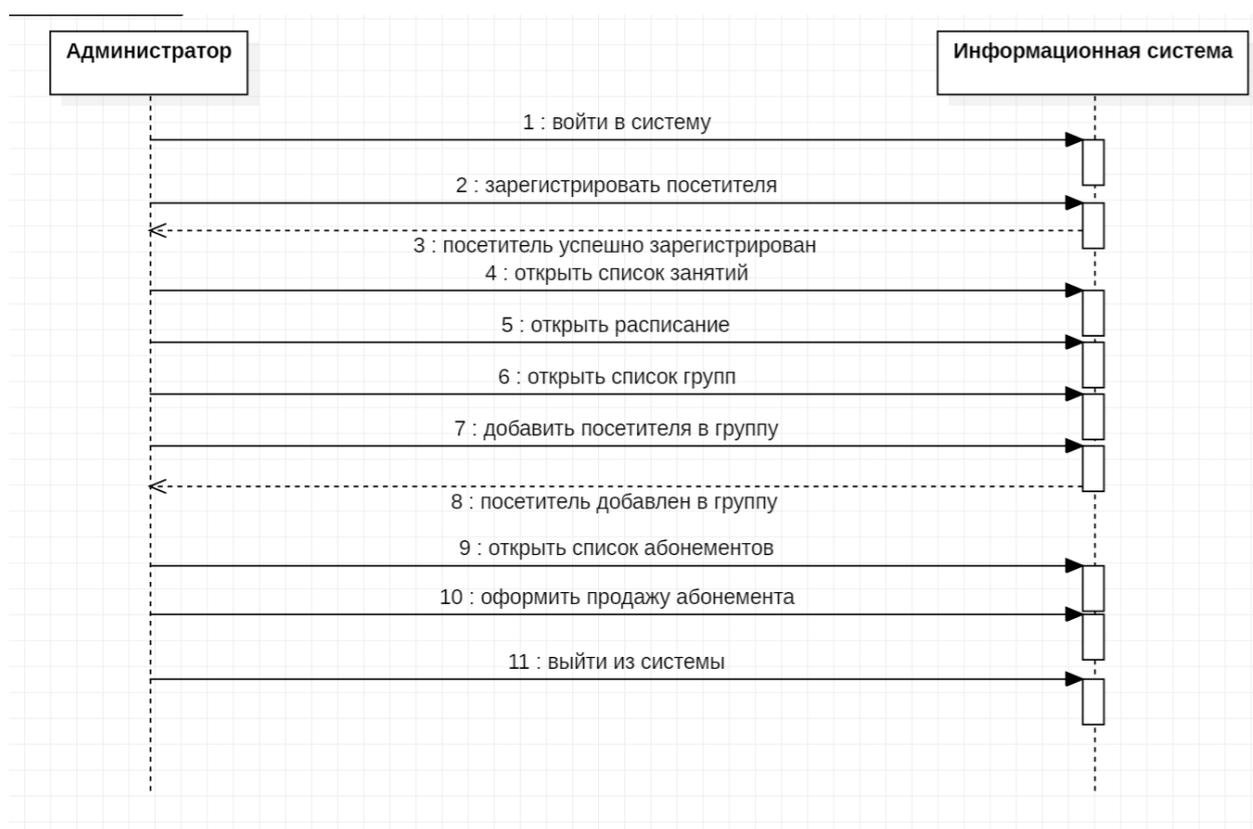


Рисунок 18 – Описание действий администратора в информационной системе

Для проектирования интерфейса пользователя можно выделить основные кнопки навигации, которые будут осуществлять переходы к данным и формам:

- информация о клиентах;
- данные об абонеентах;
- информация о расписании;
- информация о занятиях;
- форма для регистрации новых клиентов.

3.2 Структурная схема разрабатываемой системы

Для проектирования и разработки также требуется выбрать методологию, по которой команда разработчиков или один программист будет выстраивать свою работу.

DevOps – методология активного взаимодействия специалистов по разработке со специалистами по информационно-технологическому обслуживанию и взаимная интеграция их рабочих процессов друг в друга для обеспечения качества продукта [14].

Манифест гибкой разработки звучит так:

- люди важнее инструментов;
- качество продукции важнее документации;
- взаимодействие с заказчиком важнее контракта;
- готовность к изменениям важнее установленного плана.

После сравнения нескольких методологий для разработки АИС администратора физкультурно-оздоровительного центра выбрали именно методологию Канбан, так как в этой методологии нет ограничений по времени либо они очень широкие, еще есть немаловажный плюс это доработка приложения на любом из этапов, после того как основная часть приложения уже была разработана ранее, добавить несколько разделов и кнопок не занимает много времени, соответственно при реализации базы данных эти дополнения нужно учесть. Корректировки можно выполнить в любое время на этапе разработки, вернуться назад доработать и идти дальше. Есть цель, которую мы хотим достигнуть, поэтому разбили ее на задачи и последовательно их выполняли [15].

В ходе выполнения работы было поставлено несколько задач: проектирование информационной системы, создание пользовательского интерфейса, реализация базы данных и кодирование логики приложения. Сначала спроектируем базу данных и заполним ее данными, которые будут

там хранится, в дальнейшем разработаем графический интерфейс пользователя, после чего будет представление того, как будет выглядеть приложение со стороны пользователя. После создания базы и интерфейса нужно написать код, связывающий пользовательский интерфейс и данные в базе, по запросу пользователя из базы данных брать нужную информацию и выдавать результат в виде отчета на экран [16].

Авторизация пользователя в системе происходит по средствам ввода логина и пароля. После успешной авторизации пользователь входит в систему под своими данными и может осуществлять операции.

Рассмотрим более подробно процесс регистрации нового клиента в автоматизированной информационной системе, с помощью построения алгоритма и представления в виде блок-схемы процесса.

Последовательность функции «Регистрация нового клиента»:

- открыть форму заполнения заявки;
- заполнить поля: код клиента, фамилия, имя, отчество, телефон;
- нажать кнопку «ОК»;
- при пустом поле выдать сообщение (Поле не заполнено!);
- после заполнении всех полей добавлять запись в БД «Клиенты»;
- после окончания ввода нажать «Сохранить».

Блок-схема алгоритма функции «Добавить нового клиента» представлен на рисунке 19.

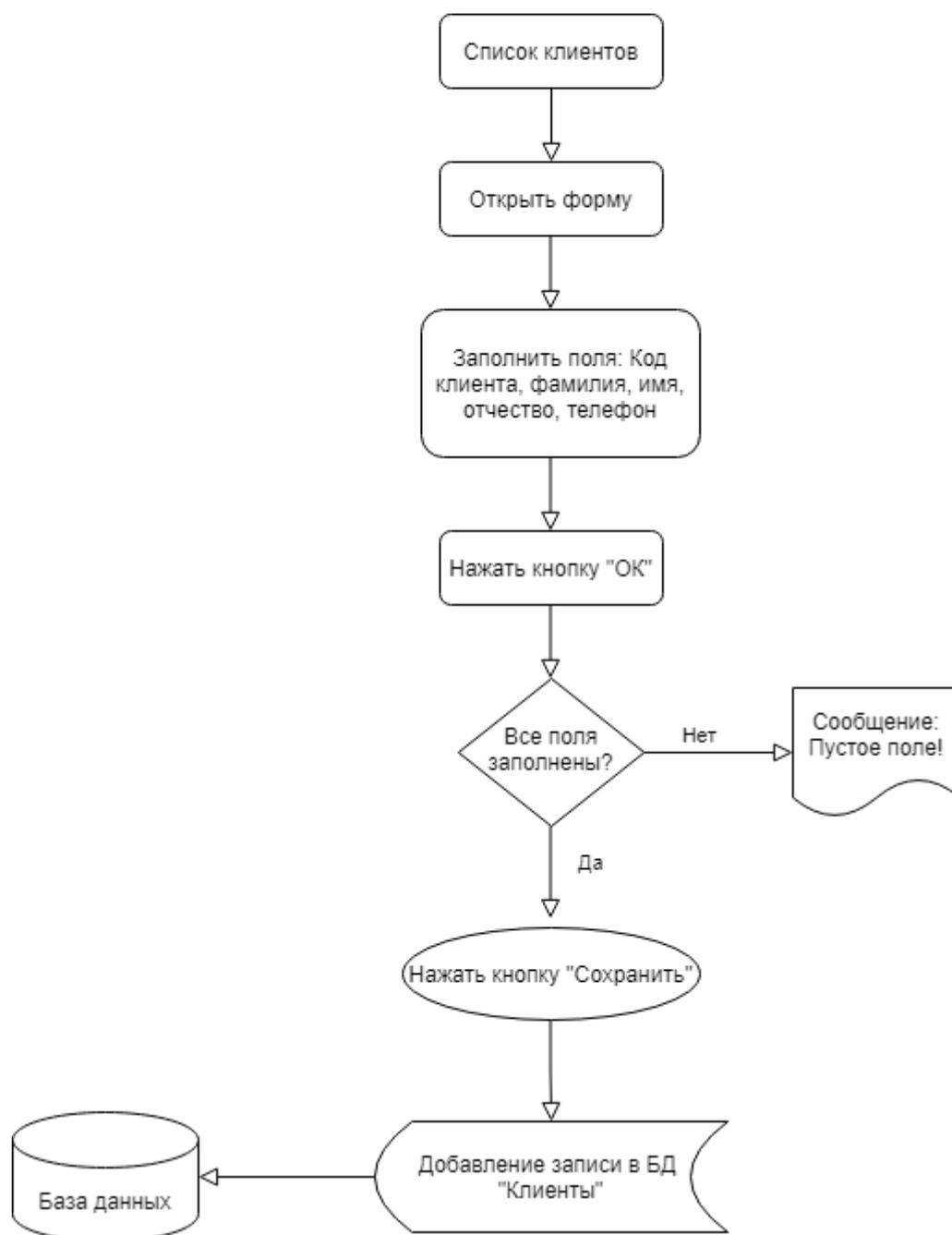


Рисунок 19 – Блок-схема алгоритма функции «Добавить нового клиента»

Чтобы начать работу необходимо установить программное обеспечение, для этого нужно перейти по адресу visualstudio.microsoft.com [17] и скачать установочный файл.

После загрузки установочного файла, его нужно запустить, после выйдет окно с установкой программы, здесь необходимо нажать кнопку

«Install» и программа начнет устанавливаться на диск C в папку Program Files [18].

3.3 Описание программных модулей

После запуска приложения предлагают выбрать пакеты инструментов необходимые для разработки.

Нам потребуется разработка классических приложений .Net, выбираем и устанавливаем пакет инструментов.

Назвать проект можно как угодно, обычно называют так, чтобы в дальнейшем было понятно, о чем он. После всех настроек для удобной работы с графическим интерфейсом будем использовать удобный паттерн MVVM [19].

MVVM – это паттерн разработки, позволяющий разделить приложение на три функциональные части:

На рисунке 20 – показана схема взаимодействия данных.

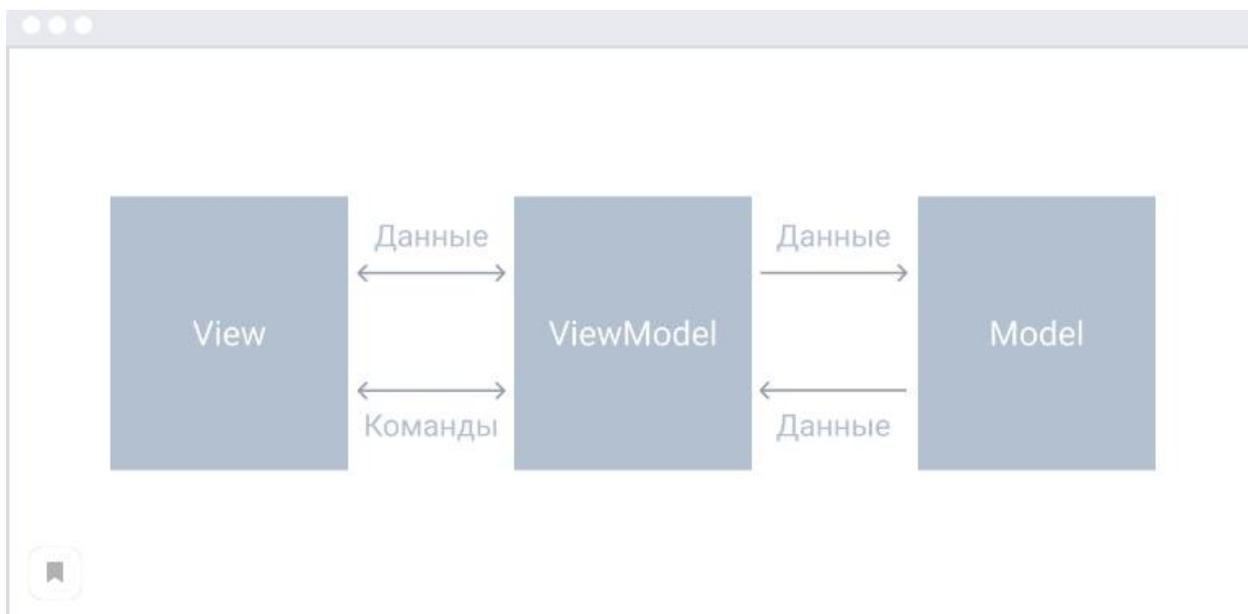


Рисунок 20 – Взаимодействие данных при использовании MVVM

Также необходимо установить базу данных, для хранения данных была выбрана MS SQL, для установки нужно перейти по ссылке www.microsoft.com и зайти в продукты, там необходимо выбрать SQL Server [20], разработчики предлагают несколько вариантов приложений.

Выберем SQL Server в локальной среде и установим бесплатную версию базы данных [21].

После установки требуется настроить имя сервера и задать логин и пароль для соединения базы данных, по окончании настроек после запуска базы данных окно соединения с сервером.

Так как у нас создана локальная база, при внедрении приложения на предприятии потребуется развернуть базу данных на сервере для того, чтобы к ней могли обращаться как администратор, так и системный администратор.

Развернуть базу данных можно разными способами, рассмотрим пример развертывания с помощью Docker.

Docker – это платформа с открытым исходным кодом, написанная на Go, которая является высокопроизводительным языком от Google. Go дает экстремальную переносимость Docker, потому что скомпилированное приложение не полагается на внешние зависимости [22].

Docker позволяет осуществлять администрирование в изолированных системах. Docker не является механизмом виртуализации, потому что он использует общие ресурсы ячеек библиотек (хоста и контейнера), а не собственные ресурсы. В виртуализации есть полная операционная система, а в Docker есть только библиотеки ядра - в Windows Nano Server [23].

Чтобы запустить SQL Server в контейнере Docker на Windows Server и Windows 10, можно выполнить все это в PowerShell.

Прежде чем начать, создавать свою машину, виртуальную или нет, нужно установить Windows Server 2016, а затем последовательно следовать коду PowerShell в режиме администратора («Запуск от имени администратора»).

После открытия окна командной строки PowerShell, нужно установить Docker, набрав команды:

```
Install-Module DockerMsftProvider -Force;
```

```
Install-Package Docker -ProviderName DockerMsftProvider -Force.
```

После установки можно проверить установился Docker или нет и его версию, набрав команду `Docker version`.

Теперь, когда Docker успешно установлен и запущен, можно приступить к созданию контейнера SQL Server. Проверить репозиторий Microsoft в Docker: `docker search microsoft` [24].

Далее необходимо загрузить репозиторий SQL Server локально, для этого нужно набрать команду: `docker pull microsoft/mssql-server-windows-developer`.

Проверить загруженные изображения локального репозитория: `Docker images`.

После ввода пароля загружается меню базы данных, где можно создавать базу, таблицы в базе, писать SQL запросы и т.д., так же можно все это делать напрямую с Visual Studio путем прописывания кода и потом можно проверить как создалась база или таблицы в ней [25].

3.4 Схема работы информационной системы

При запуске приложения каждый пользователь вводит логин или пароль, автоматизированная информационная система выглядит как пользовательский интерфейс с кнопками, для удобства заполнения заявки на обслуживание оргтехники, рисунок 21.

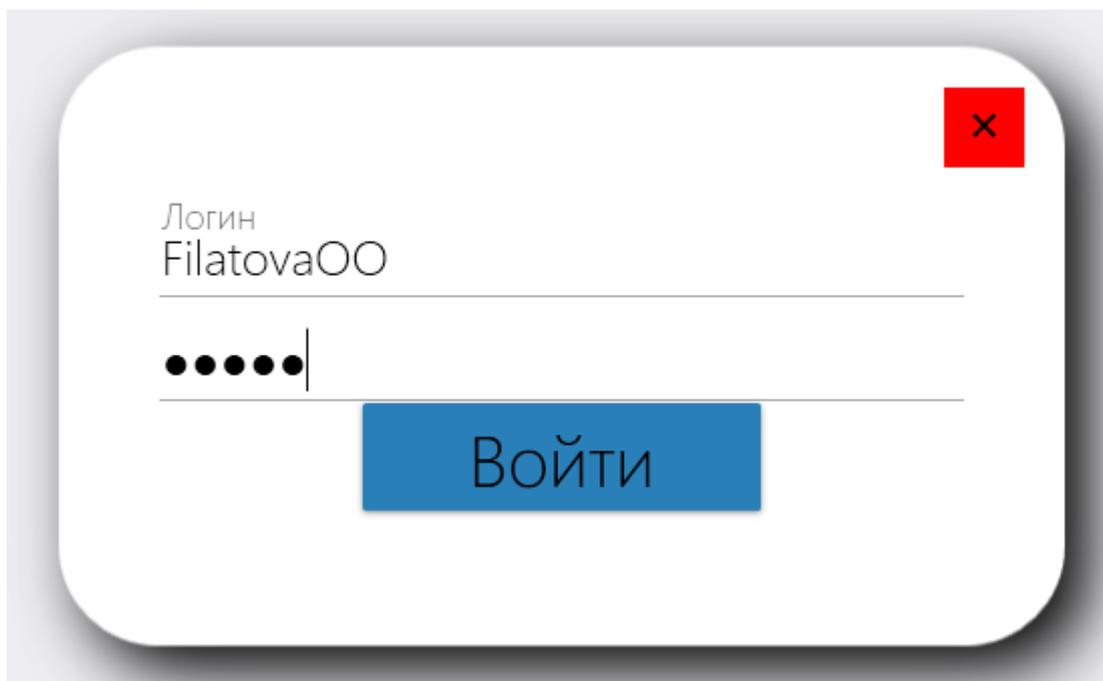


Рисунок 21 – Главное меню приложения

После ввода логина и пароля необходимо пройти авторизацию, нажав кнопку «Войти».

Для того чтобы хранить информацию необходимо заполнить базу данными, создать таблицы, которые нам необходимы для отчетов и заполнить их атрибутами.

После запуска SQL Server и ввода пароля можно создавать базу данных, для этого нужно выбрать «Database» нажать правой кнопкой и выбрать «Create» и затем «Database», выйдет окно, где пишем название нашей базы и нажимаем «ОК» [26].

Так наша база создавалась, теперь можно создавать сущности и наполнять их атрибутами.

Для дальнейшей работы в приложении, дополнительных настроек не требуется, необходимо только установить его на свой ПК и начать работать. Пользователю, который будет в нем работать не требуется дополнительного обучения или спецификации, приложение написано максимально просто для работы обычного пользователя.

После реализации графического интерфейса пользователя и создания базы данных со всей информацией.

3.5 Тестирование информационной системы

Этап тестирования обычно в финансовых затратах составляет половину расходов на создание системы. Плохо спланированное тестирование приводит к существенному увеличению сроков разработки системы и является основной причиной срывов графиков разработки.

В процессе тестирования используются данные, характерные для системы в рабочем состоянии, т.е. данные для тестирования выбираются случайным образом. План проведения испытаний должен быть составлен заранее, обычно на этапе проектирования.

Для проведения тестирования было решено провести испытания класса В. Испытания класса В выполняются независимо от группы разработки и начинаются после того, как разработчики объявляют, что изделие готово к передаче потребителю.

В каталоге, содержащем загрузочный модуль системы «ИС администратора физкультурно-оздоровительного центра», должен находиться файл «Admin.cs», содержащий в себе все необходимые методы и данные. В случае отсутствия этого файла при запуске системы на экран выводится сообщение об отсутствии необходимого файла.

В разработанной системе пользователь имеет возможности изменять данные, которые уже хранятся в базе данных, пользователь имеет возможность просматривать данные списком с обновленными данными, которые обновляются автоматически при загрузке системы.

В ходе тестирования были выявлены следующие недостатки программы:

При отсутствии файла работа с системой невозможна. Вариант исправления: при отсутствии файла Admin.cs в указанной выше папке

производить создание нового пустого файла или открыть папку Debug и запустить систему.

Система «ИС администратора физкультурно-оздоровительного центра» предназначена для выполнения функций деятельности администратора в течении рабочего дня туров клиентам.

Технические и программные средства, обеспечивающие выполнение данной программы: процессор Pentium IV или эквивалентный ему, не менее 80 Мб на жестком диске для самой программы, устройства ввода (клавиатура, мышь), устройство вывода (монитор) Программа работает под управлением операционной системы Windows 10.

Проверка программы производится в следующем порядке:

- 1) запуск системы;
- 2) проверка наличия загрузочного файла;
- 3) если работа системы закончилась успешно и появилось окно графического интерфейса, то проверка считается успешно завершённой и следует выйти из программы;
- 4) если работа программы завершилась не успешно, то следует обратиться к разработчику программы.

При получении диагностических и иных сообщений в ходе проверки программы следует обращаться к разделу «Сообщения системному программисту» данного руководства.

Система выполняет регистрацию нового клиента (посетителя), дает возможность просматривать списки данных по разным категориям, программа сама исправляет ошибки информации, если система по какой-либо причине не смогла выполниться, то пользователь не увидит окно графического интерфейса с данными, администратор системы сможет увидеть сообщение об ошибках и предпринять меры по их устранению.

3.6 Руководство пользователя

После прохождения авторизации пользователь попадает в главное меню администратора физкультурно-оздоровительного центра, рисунке 22.

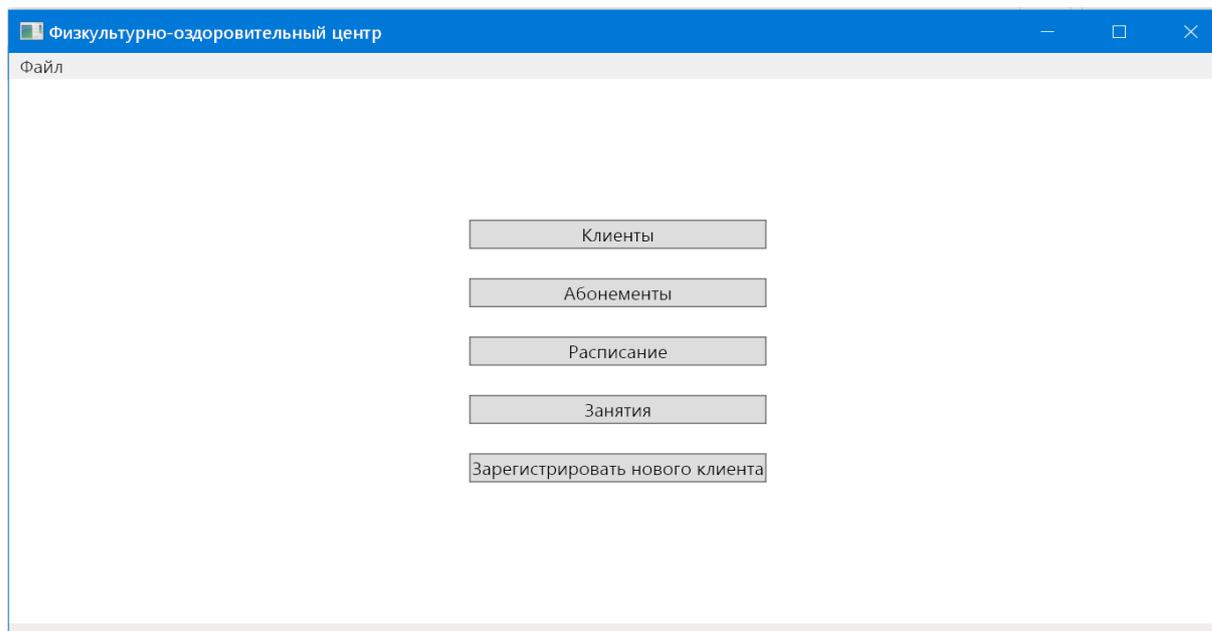
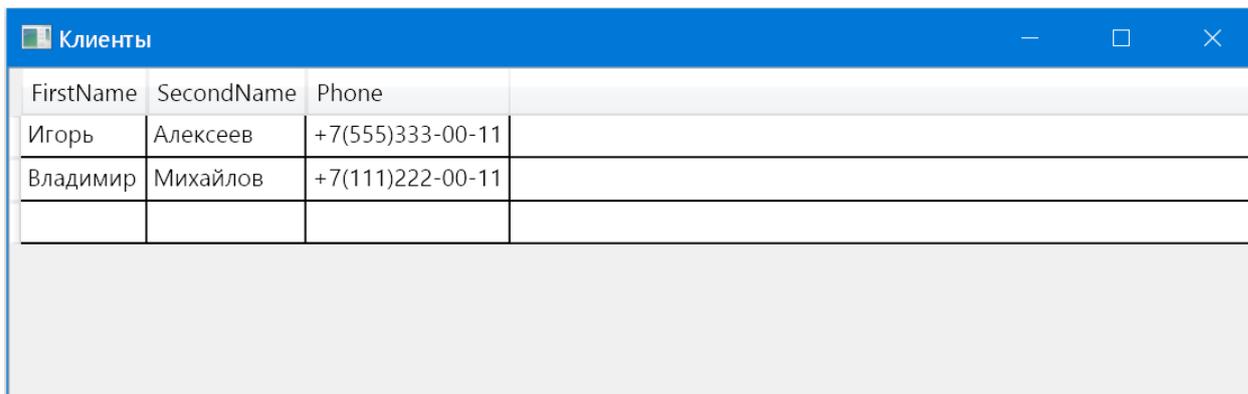


Рисунок 22 – Окно администратора

На рисунке 23 представлен скриншот таблицы «Клиенты» для редактирования справочных данных, используемых в разработанной системе [27].



FirstName	SecondName	Phone	
Игорь	Алексеев	+7(555)333-00-11	
Владимир	Михайлов	+7(111)222-00-11	

Рисунок 23 – Скриншот таблицы «Клиенты»

Администратор при посещении первого раза клиентом обязан его зарегистрировать в системе, чтобы появилась возможность записать его в группу на занятия и оформить абонемент. На рисунке 24 показана форма регистрации нового клиента.

Регистрация нового клиента	
Имя клиента	Игорь
Фамилия клиента	Алексеев
Телефон клиента	+7(555)333-00-11

Ok Cancel

Рисунок 24 – Форма регистрации нового клиента

После того как введены все данные клиента, администратор может нажать кнопку «Ok», после чего данные клиента автоматически будут добавлены в базу данных. Также у администратора есть возможность нажать кнопку «Cancel», форма будет закрыта, без сохранения, если не нажата кнопка «Ok» и с сохранением при ее нажатии [28].

Выводы по третьему разделу

В третьем разделе работы рассмотрено дерево функций, описан интерфейс информационной системы администратора физкультурно-оздоровительного центра. Используя дерево функций, появилась возможность разработать схему информационной системы с помощью диаграмм и блок-схем. Также были описаны алгоритмы выполнения некоторых функций системы, которые наиболее востребованы администратором. Для удобства использования информационной системой было написано руководство пользователя.

Заключение

В результате выполнения выпускной квалификационной работы были выполнены все поставленные задачи:

- проведен обзор предметной области и получена информация о бизнес-процессах предприятия;
- сформированы функциональные требования к автоматизированной информационной системе;
- проведен обзор аналогичных программных продуктов, решающих подобные задачи на основе чего была подтверждена актуальность разработки автоматизированной информационной системы;
- с помощью сравнительного анализа были выбраны ЯП С#, СУБД MS SQL Server и IDE Visual Studio 2019;
- разработана логическая и физическая модели базы данных;
- реализована база данных в СУБД MS SQL Server;
- реализованы необходимые функции пользовательского интерфейса, необходимые для работы администратора физкультурно-оздоровительного центра;
- разработана автоматизированная информационная система «Физкультурно-оздоровительный центр» и успешно протестирована, что показало соответствие разработанных функций предъявляемым требованиям.
- описаны алгоритмы выполнения некоторых функций системы, которые наиболее востребованы администратором.
- написано руководство пользователя.

Разработанную информационную систему можно будет внедрить в деятельность небольшого центра, а также обновлять функционал системы по мере необходимости при расширении физкультурно-оздоровительного центра.

Все задачи выполнены в полном объеме, цель работы достигнута.

Список используемой литературы

1. Бистерфельд О.А. Методология функционального моделирования IDEF0: учебно-методическое пособие / О.А. Бистерфельд; Ряз. гос. ун-т им. С.А. Есенина. — Рязань, 2008. — 48 с.
2. Гвоздева, Т.В. Проектирование информационных систем. Методы и средства функционального проектирования /Т.В. Гвоздева. М.: Лань, 2019. — 458 с.
3. Глотова, С.А. Информационные системы /С.А. Глотова. М.: Темика, 2020. — 494 с.
4. ГОСТ 34.003-90 Автоматизированные системы. Термины и определения.
5. ГОСТ 24.702-85 Единая система стандартов автоматизированных систем управления. Эффективность автоматизированных систем управления. Основные положения.
6. Гусов К. Н., Андреев А. А., Бондаренко Э. Н. Трудовое право России. Учебник для бакалавров. — М.: Проспект, 2021. 592 с.
7. Горбаченко В. И. Проектирование информационных систем с СА ERwin Modeling Suite 7.3: учебное пособие / В. И. Горбаченко, Г. Ф. Убиенных, Г. В. Бобрышева – Пенза: Изд-во ПГУ, 2019. — 154 с
8. Методология Kanban: доски, принципы и возможности управления. [Электронный ресурс]. — Режим доступа: <https://skillbox.ru/media/>(дата обращения: 13.10.2022)
9. Онлайн курс обучения программированию: методологии разработки. [Электронный ресурс]. — Режим доступа: <https://javarush.ru/groups/posts/647-metodologii-razrabotki-po> (дата обращения: 13.10.2022)
10. Отличия, достоинства и недостатки базы данных PostgreSQL: что такое PostgreSQL. [Электронный ресурс]. — Режим доступа: <https://oracle-patches.com/common/3214-что-такое-postgresql> (дата обращения: 13.10.2022)

11. Паттерн MVVM: сто такое MVVM. [Электронный ресурс]. – Режим доступа: <https://skillbox.ru/media/code/> (дата обращения: 13.10.2022)
12. Семенова О.В. Информационные системы и программирование / О.В. Семенова. – М.: Инфра-М, 2020. – 214 с.
13. С GUI на C#: Кроссплатформенное приложение [Электронный ресурс]. - Режим доступа: https://skillbox.ru/media/code/ne_windows_ (дата обращения: 13.10.2022)
14. Черушева Т. В. Проектирование программного обеспечения: учеб. пособие / Т. В. Черушева. – Пенза: Изд-во ПГУ, 2018. – 172 с.
15. Физкультурно-оздоровительная деятельность: организационные мероприятия, методы обеспечения, основные функции и планы развития [Электронный ресурс]. Режим доступа: <https://fb.ru.turbopages.org/fb.ru/s/article/445283/fizkulturno-ozdorovitel'naya-deyatelnost-organizatsionnyie-meropriyatiya-metodyi-obespecheniya-osnovnyie-funktsii-i-plany-i-razvitiya> (дата обращения 13.10.2022).
16. Хорев П.Б. Программно-аппаратная защита информации / П.Б. Хорев. – М.: Форум, 2019. – 352 с.
17. Худяков Г.А., Проектирование информационных систем / Г.А. Худяков. Учебное пособие, М.– Омега–Л, 2021. – 432 с.
18. Docker: Развертывание контейнеров Docker в SQL Server [Электронный ресурс]. - Режим доступа: <https://docs.microsoft.com/ru-ru/sql/linux/sql-server-linux-docker-container-deployment?view=sql-server-ver15&pivots=cs1-bash> (дата обращения: 13.10.2022)
19. Firebase [Электронный ресурс]. Режим доступа: <https://a2is.ru/catalog/programmy-dlya-tantsevalnykh-studij-i-shkol-tantsev/fitbase> (дата обращения 13.10.2022).
20. GeekBrains – Язык программирования C#. [Электронный ресурс]. - Режим доступа: <https://geekbrains.ru/posts/yazyk-programmirovaniya-c-sharp-istoriya-spezifika-mesto-na-rynke> (дата обращения: 13.10.2022)

21. Helpiks.org: Достоинства и недостатки языка [Электронный ресурс]. - Режим доступа: <https://helpiks.org/6-21879.html> (дата обращения: 13.10.2022)
22. Mobifitness [Электронный ресурс]. URL: <https://a2is.ru/catalog/crm-dlya-upravleniya-fitness-tsentrom/mobifitness> (дата обращения 13.10.2022).
23. Sport Priority [Электронный ресурс]. URL: <https://a2is.ru/catalog/crm-dlya-upravleniya-fitness-tsentrom/sport-priority> (дата обращения 13.10.2022).
24. SoftClipper: что такое SQL Server. [Электронный ресурс]. – Режим доступа: <https://softclipper.net/foxpro-i-sql/sravnenie-baz-dannykh-microsoft-sql-server-i-microsoft-visual-foxpro.html> (дата обращения: 13.10.2022)
25. SQL Server: руководство по SQL Server. [Электронный ресурс]. – Режим доступа: <https://docs.microsoft.com/ru-ru/sql/sql-server/tutorials-for-sql-server-2016?view=sql-server-ver15> (дата обращения: 13.10.2022)
26. SQL Server: программное обеспечение. [Электронный ресурс]. – Режим доступа: <https://www.microsoft.com/ru-ru/sql-server/sql-server-downloads> (дата обращения: 13.10.2022)
27. Shell: Программирование на языке командного интерпретатора [Электронный ресурс]. - Режим доступа: <https://www.lissyara.su/doc/> (дата обращения: 13.10.2022)
28. Visual Studio 2019: программное обеспечение. [Электронный ресурс]. – Режим доступа: <https://visualstudio.microsoft.com/ru/vs/> (дата обращения: 13.10.2022)