

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное  
учреждение высшего образования  
«Южно-Уральский государственный университет»  
(национальный исследовательский университет)  
Высшая школа экономики и управления  
Кафедра «Информационные технологии в экономике»

РАБОТА ПРОВЕРЕНА

Рецензент, \_\_\_\_\_  
\_\_\_\_\_/\_\_\_\_\_/\_\_\_\_\_  
« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

ДОПУСТИТЬ К ЗАЩИТЕ

Зав. кафедрой, д.т.н., с.н.с.  
\_\_\_\_\_/ Б.М. Суховилов /  
« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

Разработка информационной системы для стоматологической клиники

ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА  
ЮУрГУ – 09.04.03.2021.039. ВКР

Руководитель, должность

Е.М. Саргасов, доц. каф. ИТЭ

\_\_\_\_\_/\_\_\_\_\_  
« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

Автор

студент группы ЭиУ-221

/ А.В. Найко /

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

Нормоконтролер, доц. каф. ИТЭ

Е.Н. Горных

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

Челябинск 2021



## АННОТАЦИЯ

Найко А.В. Разработка информационной системы для стоматологической клиники – Челябинск: ЮУрГУ, ЭУ-221, 72 с., 28 ил., 17 табл., библиогр. список – 21 наим., 6 прил.

Разработана информационная система для автоматизации бизнес – процессов в стоматологической клинике. В работе проанализированы существующие программные решения, выявлены их основные преимущества и недостатки.

Для достижения цели был проведен анализ требований к разрабатываемому продукту. Проанализирована литература, интернет – источники, посвященные автоматизации бизнес – процессов, типовым программным решениям для автоматизации бизнес – процессов в стоматологии, представленным на рынке программного обеспечения, а также нормативная документация.

С помощью методологии DFD и IDEF0 созданы графические модели прикладной задачи, описаны основные объекты и связи между ними. Наглядно представлено взаимодействие базы данных с приложением.

Разработано интуитивно понятное и простое в восприятии приложение, рассчитанное для малых предприятий. Решение содержит ключевую информацию по инвестиционной деятельности для принятия решения.

Рассчитаны затраты на разработку, обоснована экономическая эффективность проекта.

# ОГЛАВЛЕНИЕ

АННОТАЦИЯ .....	10
ОГЛАВЛЕНИЕ .....	11
ВВЕДЕНИЕ .....	8
1 ПРОБЛЕМЫ И ТЕНДЕНЦИИ РАЗВИТИЯ ИНФОРМАЦИОННЫХ СИСТЕМ В ОБЛАСТИ СТОМАТОЛОГИИ.....	10
1.1 Анализ источников по теме работы .....	10
1.2 Сущность и виды процессов в сфере оказания стоматологических услуг .	13
1.3 Обоснование необходимости и цели использования программного обеспечения для автоматизации работы стоматологической клиники.....	15
1.4 Анализ типовых программных решений для автоматизации стоматологической клиники.....	16
1.5 Тенденции развития информационных систем в стоматологии.....	21
Выводы по первому разделу.....	22
2 ПРОЕКТНАЯ ЧАСТЬ .....	23
2.1 Выбор инструментария решения задачи.....	23
2.3 Характеристики деятельности стоматологической клиники.....	25
2.4 Обоснование проектных решений.....	28
2.3 Структура базы данных .....	31
2.4 Создание схемы членства в SQL Server .....	36
2. 5 Интерфейс программы .....	45
Выводы по второму разделу.....	56
3 ОРГАНИЗАЦИОННО-ЭКОНОМИЧЕСКАЯ ЧАСТЬ.....	57
3.1 Затраты на материалы .....	57
3.2 Затраты на заработную плату .....	57
3.3 Затраты на амортизацию .....	58
3.4 Прочие затраты.....	59
Вывод по разделу три .....	60
ЗАКЛЮЧЕНИЕ .....	61

БИБЛИОГРАФИЧЕСКИЙ СПИСОК .....	63
ПРИЛОЖЕНИЕ А КОД КЛАССА LOGIN .....	66
ПРИЛОЖЕНИЕ Б КОД ФУНКЦИИ POPULATEGRIDVIEW() .....	67
ПРИЛОЖЕНИЕ В КОД СОБЫТИЯ ROWCOMMAND .....	68
ПРИЛОЖЕНИЕ Г КОД СОБЫТИЯ ROWUPDATING .....	69
ПРИЛОЖЕНИЕ Д КОД СОБЫТИЯ ROWDELETING .....	70
ПРИЛОЖЕНИЕ Е КОД СОБЫТИЯ CLICK .....	71

## **ВВЕДЕНИЕ**

За последние десять лет в нашей стране быстрыми темпами начал развиваться малый и средний бизнес. Расширение бизнеса требует четкого и своевременного контроля, который можно получить, автоматизируя бизнес-процессы.

Автоматизация бизнес процессов стоматологических клиник влияет на качество и скорость их выполнения, а значит в целом на качество предоставляемых стоматологических услуг и успешность предприятия в целом.

Автоматизация позволит эффективно использовать трудовые ресурсы, повысить эффективность управленческой деятельности, ведения учета пациентов, а также повысить качество предоставляемых услуг за счёт сокращения времени на оформлении документации. Кроме того, автоматизация позволит рационализировать производственную деятельность, а, следовательно, повысить экономические показатели организации. Одной из самых сложных задач стоматологической клиники, является сбор, обработка, хранение и предоставление данных о пациентах, результатах приёмов. Учет пациентов включает в себя хранение личных данных, информации об оказанных услугах, их стоимость, а также договоров, заключенных с клиентами клиники. На основании учета пациентов формируется общая картина стоматологии.

Предметом исследования является деятельность стоматологической клиники, а в частности процессы учета пациентов, материалов, ведения медицинских карт и т.д.

Цель работы – разработать информационную систему для стоматологической клиники.

В соответствии с поставленной целью в работе определены следующие задачи:

- провести анализ предметной области;
- разработать проект системы;

- разработать модель потока данных (DFD);
- построить схему базы данных;
- реализовать систему на платформе ASP.NET;
- провести опытную эксплуатацию;
- рассчитать экономическую эффективность.

# 1 ПРОБЛЕМЫ И ТЕНДЕНЦИИ РАЗВИТИЯ ИНФОРМАЦИОННЫХ СИСТЕМ В ОБЛАСТИ СТОМАТОЛОГИИ

## 1.1 Анализ источников по теме работы

В настоящее время информационные технологии, внедряемые с целью автоматизации, используются во всех областях деятельности человека. Глобальная задача автоматизации – повышение эффективности компании: повышение внутренней управляемости и устойчивости к внешним воздействиям. Более подробно о задачах автоматизации пишет Курьянов С. С. в своей статье «Актуальные задачи и пути автоматизации управления бизнес-процессами» [9]. Автор считает, что решение этих задач заключается во внедрении систем автоматизации всех бизнес-процессов предприятия.

Важным преимуществом таких систем будет гибкость, что позволит в точности повторить всю структуру бизнес процесса. Дополнительно по этой теме была рассмотрена статья Лифшиц И. М. «"По зубам" ли дантистам системы автоматизации?», где автор анализирует состояние рынка IT– стоматологии, и даёт оценку тенденции развития автоматизации современных стоматологических клиник на ближайшие 2– 3 года [21]. Автор считает, что ключевым фактором выбора систем станут уровень и качество предоставляемой поддержки, репутация компании-разработчика и гибкость в настройке функционала программного продукта.

Внедрение систем, автоматизирующих бизнес-процессы, как и серьёзные преобразования на предприятии – это сложный и, зачастую, болезненный процесс, который приводит к ряду проблем. На данный момент, большинство из них, достаточно хорошо изучены, формализованы и имеют эффективные методы решения. На эту тему пишут Филимонов В. Ю., Шурыгин И. С., Тынченко В. С. в статье «Проблемы при внедрении процессного подхода к автоматизации бизнеса и механизмы их решения», приводя проблемы и задачи, которые



возникают в большинстве случаев при внедрении, и рекомендации для их решения [29]. Изучение и подготовка к этим проблемам заблаговременно, позволит облегчить процесс внедрения и повысить эффективность последующего использования системы.

Основные проблемы и задачи, которые требуют особого внимания:

- потребность в полной или частичной реорганизации структуры предприятия;
- потребность в изменениях технологии бизнеса;
- сопротивление сотрудников предприятия переходу/внедрению новой информационной системы (ИС);
- увеличение нагрузки на сотрудников на этапе внедрения системы;
- потребность в создании группы внедрения и сопровождения системы.

Особое внимание уделяет Верников В. В. сопротивлению внедрению в статье «Руководителю предприятия. Внедрение системы автоматизации, основные проблемы и задачи» [32]. Автор считает, что это одна из самых частых причин, которая способна сорвать или существенно затянуть процесс. Он рекомендует, создать у сотрудников ощущение неизбежности внедрения системы, а также издавать соответствующие приказы, распоряжения или другие нормативные документы.

Внедряемая ИС должна улучшать информационный обмен между отделами предприятия, сокращать время разработки и согласования документооборота. В статье «Автоматизация бизнес-процессов малого предприятия» [5] Мололкина О. Л. на примере малого предприятия по продаже цветов и сопутствующей продукции, выделены плюсы и минусы, с которыми можно столкнуться при выборе программного решения; автор рассказывает, в каких случаях стоит отдавать предпочтение разработке нового продукта.

В наше время системы, внедряемые в стоматологических клиниках, позволяют не только вести электронную картотеку, но и содержат в себе справочники по известным нозологическим формам стоматологических

заболеваний, имеют схемы дифференциальной диагностики и алгоритмов лечения. В статье «Создание автоматизированной компьютерной системы для информационной поддержки врача-стоматолога» Брагин А. В., Мирошниченко В. В. и Орлова Е. С. пишут о методике разработки подобных автоматизированных систем для стоматологии, которые позволят повысить качество медицинской помощи [33]. В статье «Автоматизация управления стоматологической организацией с использованием методов проектного и процессного управления» Садовский В. В. и Сапрыкина А. В. считают иначе, утверждая, что стоит уделять внимание разработке ИС следующего поколения, которые будут создаваться на основе полнофункциональных систем ERP-класса, интегрированных систем для управления основными внутренними и внешними ресурсами предприятия и обеспечивать автоматизацию не только лечебного процесса, но и комплексно решать другие задачи управления стоматологической клиникой [6]. Авторы рекомендуют при создании ИС следовать принципам проектного и процессного управления.

Процессный подход к управлению позволяет рассматривать предприятие, как множество связанных между собой бизнес-процессов, а не объединение разрозненных функций [30]. Каждый отдельно взятый бизнес-процесс – это последовательность операций, нацеленных на достижение результата: все сотрудники предприятия знают, какую работу, в какой срок и какого качества им необходимо выполнять, чтобы бизнес-процесс, в котором он участвует, привел к желаемому результату. Исходя из этого возникла необходимость в изучении нормативной документации, где содержится информация о задачах, которые ставят перед собой стоматологические клики, информация и возможных предоставляемых услугах и порядок их проведения, должностные инструкции стоматолога [13], администратора стоматологической клиники [8] и медицинской сестры стоматологического кабинета [14].

В процессе разработки возникает ряд вопросов, связанных с технологией разработки программного обеспечения, которые можно решить, обратившись к

ГОСТ Р ИСО/МЭК 12207-2010 «Информационная технология (ИТ). Системная и программная инженерия. Процессы жизненного цикла программных средств» [11]. В документе указано, что одним из первых и самых важных этапов разработки является написание технического задания (ТЗ). Получить универсальную форму и необходимые пояснения по составлению ТЗ можно, изучив ГОСТ 34.602-1989 «Информационная технология. Комплекс стандартов на автоматизированные системы. Техническое задание на создание автоматизированной системы» [12]. Данный стандарт позволит построить и оформить техническое задание на разработку программных продуктов или программных изделий для вычислительных машин, независимо от области их применения.

## **1.2 Сущность и виды процессов в сфере оказания стоматологических услуг**

Стоматологическая клиника, как и любые предприятия, – это совокупность бизнес-процессов (производственных и коммерческих операций), направленных на получение прибыли за счёт создания ценностей для потребителей.

Все процессы можно разделить на три группы: обеспечивающие, основные и процессы управления. На рисунке 1 показаны процессы в стоматологической клинике для каждой группы.



Рисунок 1 – Процессы в стоматологической клинике

Важную роль в эффективной работе клиники играет правильно организация регистратуры, основные функции которой – учёт пациентов и ведение картотеки. Отлаженная система учёта позволяет быстро найти необходимую информацию и воспользоваться ею, например, информация о ранее поставленном диагнозе, а также выдать её по форме организации по требованию пациента. Основные процессы стоматологической клиники сопровождаются заполнением карты об итогах приёма. Многие медицинские организации пользуются единой формой карты 043/у.

Для платных стоматологических клиник важно проработать один из обеспечивающих процессов – управление документацией, так как с каждым пациентом, до начала приёма, заключается договор на оказание медицинских услуг, который создаётся в двух экземплярах – для организации и для клиента. Наличие отлаженного процесса учёта/закупки материала, а также технического оборудования, позволит не потерять клиентов, оказав услуг здесь и сейчас, либо предоставив точную информацию о дате возможно приёма/исполнения.

### **1.3 Обоснование необходимости и цели использования программного обеспечения для автоматизации работы стоматологической клиники**

Последние десятилетия заметно высокое использование информационных технологий в бизнесе: программные продукты (ПП) для финансового учета, автоматизированные системы для проектирования и системы планирования производства и т.п. В настоящее время практически не осталось организаций, не использующих ПП для автоматизации процессов.

Деятельность стоматологической клиники базируется на хранении информации и ведении различной документации. Иногда возникает необходимость в формировании отчётов/списков с информацией о пациентах, их диагнозах, приёмах и использованных медикаментозных средств. Поиск и обработка такого объёма информации может отнимать довольно большой промежуток времени.

Зачастую, состояние информационной системы стоматологических клиник одинаковое и можно выделить основные недостатки, подлежащие устранению:

- снижение качества предоставляемых услуг за счёт долгой обработки информации;
- зачастую, неудачное планирование работы персонала, приема пациентов, закупки стоматологических и расходных материалов;
- отсутствие единой информационной базы, что не даёт качественно взаимодействовать подразделениям в стоматологии и сотрудникам;
- отсутствие унификации и систематизации данных о пациентах, персонале и оборудовании клиники, проведенных работах, использованных материалах, расчетах с клиентами;
- отсутствие возможности своевременного и быстрого анализа и мониторинга деятельности клиники ее руководством и менеджментом для

осуществления дальнейшего перспективного планирования и повышения эффективности бизнеса.

Исходя из выделенных недостатков и многообразия бизнес-процессов в данной сфере важно большое внимание уделить программным решениям, которые бы их автоматизировали.

Современные системы в стоматологических клиниках не только позволяют вести учёт и хранить информацию о деятельности организации, но и помогают сотрудникам реализовывать основные процессы. Так, стоматологам при заполнении медицинской карты во время приёма, система подсказывает возможный диагноз, а также предлагает список препаратов для устранения симптомов.

Используемое программное обеспечение должно давать возможность вести справочники, содержащие информацию о клиентах, контрагентах, сотрудниках и др.

Формирование разного типа отчетностей поможет своевременно выявить провалы и узкие места в процессах предприятия и способствовать их устранению.

Необходимо также учитывать, что в будущем может потребоваться расширение функционала информационной системы, поэтому важно, чтобы программа имела возможность доработки и надстройки необходимого функционала.

#### **1.4 Анализ типовых программных решений для автоматизации стоматологической клиники**

Рынок программного обеспечения для стоматологических клиник в настоящее время является наиболее бурно развивающимся. Одни предлагают простой функционал с интуитивно понятным дизайном, другие стараются выделиться и предложить новые решения для автоматизации нетиповых процессов.

Чтобы успешно проанализировать и дать оценку программным продуктам были выделены следующие обязательные критерии:

- возможность дорабатывать систему под нужды предприятия;
- реализация справочников или иных средств для структуризации данных;
- наличие стандартных отчетов, характерных для данной сферы или возможность их создания;
- простота и удобство интерфейса;
- возможность разграничивать функционал системы для каждой категории пользователей;
- реализована служба технической поддержки.

«ClinicIQ» [35]

Разработчик: ClinicIQ.

Стоимость: 2600 рублей/мес. для одного врача и 6900 рублей/мес. для 5 врачей.

О программе: «ClinicIQ» – онлайн сервис для медицинских центров и для стоматологий.

Функционал:

- запись на приём;
- обзвон перед визитом и отправка смс-напоминаний;
- распечатка договоров и согласий, расчёты;
- возможность назначать и контролировать повторные посещения пациентов, напоминания о визите;
- создание расписания работы клиники;
- ведение базы пациентов и работа с их электронными картами, импорт базы;
- формирование счетов для расчета с пациентами, оплаты от пациентов;
- создание профилей сотрудников, задание мотивации для расчета заработной платы;

- организация и ведение прайс-листа клиники;
- разграничение прав доступа сотрудникам клиники согласно правилам

безопасности;

- наличие готовых стандартных отчетов.

Преимущества:

- простой и удобный интерфейс;
- бесплатная установка и техническая поддержка;
- доступ к данным в режиме 24/7;
- возможность выгрузки всех данных клиники в MS Excel;
- резервирование данных каждые 15 мин;
- наличие мобильного приложения для сотрудников клиники с

актуальным расписанием.

Недостатки:

- необходимо хорошее и стабильное подключение к интернету;
- при неоплате подписки происходит потеря доступа к данным;
- закрытый код;
- нет возможности дорабатывать систему под нужды предприятия.

«DentalBase» [36]

Разработчик: Андриянов Валерий Владимирович.

Цена: 3600 рублей/мес.

О программе: «DentalBase» представляет собой облачный сервис для стоматологий по ведению расписания, анкет пациентов, финансов, проделанной работе по пациенту, и т.д.

Функционал:

- ведение базы клиентов;
- мониторинг эффективности персонала;
- создание различных отчетов;
- ведение и сохранение истории взаимодействия с клиентом;



- организация рассылок;
- экспорт/импорт данных из других систем.

Преимущества:

- облачный сервис без ограничений;
- лёгкая настройка;
- система построена по отдельной функциональности под каждого пользователя.

Недостатки:

- необходимо хорошее и стабильное подключение к интернету;
- при неоплате подписки происходит потеря доступа к данным;
- закрытый код;
- нет возможности дорабатывать систему под нужды предприятия;
- высокая цена за одно место, нет скидок при покупке нескольких рабочих мест.

«БИТ.Стоматология» [18]

Разработчик: Первый Бит.

Цена: 3600 рублей/мес. при оплате 6 месяцев аренды.

О программе: «БИТ.Стоматология» представляет собой комплексное решение для эффективного управления стоматологической клиникой.

Функционал:

- ведение клиентской базы;
- учет взаимодействий со страховыми организациями;
- формирование реестра оказанных услуг;
- ведение взаиморасчётов с клиентами, учет оплат, контроль состояния взаиморасчётов, разделение по видам оплат;
- ведение электронной медицинской карты пациента;
- ведение зубных карт;

- хранение фотографий, рентгеновских снимков и других прикрепленных файлов пациента;

- создание отчетов, а также наличие стандартных отчетов для стоматологий;

- складской учет материалов;
- расчет заработной платы сотрудников;
- разделение прав доступа к информации.

Преимущества:

- простота и удобство работы;
- интеграции с оборудованием, сайтом, мобильными приложениями;
- SMS-информирование, ведение истории взаимоотношений, контактной информации, скидки, акции, сертификаты и др.;

- открытый код программы;
- возможность самостоятельной доработки без участия разработчика;
- регулярное обновление на основе анализа нововведений в законодательстве и запросов клиентов;

- оперативная линия консультаций разработчика.

Недостатки:

- необходимо хорошее и стабильное подключение к интернету, если выбор отдаётся в пользу облачного сервиса;

- при неоплате подписки происходит потеря доступа к данным; • высокая цена за одно место.

Сводный отчет, согласно выделенным критериям отражен в таблице 1.

Таблица 1 – Сводная таблица по анализу существующих разработок

<b>Наименование/Характеристики</b>	<b>Бит.Стоматология</b>	<b>DentalBase</b>	<b>ClinicIQ</b>
Возможность дорабатывать систему под нужды предприятия.	+	-	-
Реализована служба технической поддержки.	+	+	+
Возможность разграничивать функционал системы для каждой категории пользователей.	+	+	-

Простота и удобство интерфейса.	+	+	+
Наличие стандартных отчетов, характерных для данной сферы и возможность их создания.	+	+	Реализованы только стандартные отчёты

### 1.5 Тенденции развития информационных систем в стоматологии

Проведя анализ популярных программных решений для стоматологических клиник, можно выделить следующие тенденции [26]:

- добавление большого многообразия справочников, содержащих информацию практически по всем видам деятельности стоматологических клиник;
- реализация технологий, которые обеспечивали бы интерактивный доступ пользователей системы к информационным ресурсам;
- включение в информационные системы стоматологических клиник экспертных систем, систем машинного перевода и других технологических средств;
- увеличение сложности информационных продуктов, услуг и технологий их использования;
- создание различных решений с применением сервисов облачных технологий.

Современные информационно-компьютерные технологии в сфере стоматологии должны способствовать оптимизации всех процессов, в том числе учет материально-технической базы, решать вопросы планирования мероприятий по повышению квалификации кадров и улучшать качества оказываемой медицинской помощи [28].

Автоматизация учета материально-технической базы должна обеспечить возможность:

- выявлять недостающее оборудование в стоматологической клинике, исходя из указанных параметров;

- составлять список недостающего оборудования, который можно конвертировать в различные форматы.

В системах должна быть предусмотрена возможность поддержки принятия управленческого решения в отношении формирования потоков специалистов, нуждающихся в повышении квалификации или переподготовки, присвоении (подтверждении) категории.

Реализация систем, отслеживающих качественные и количественные показатели работы стоматологов, позволит:

- вести ежедневный учет проведенных работ и оказанных услуг по различным срезам;
- создавать индивидуальный и суммарный отчет за любой промежуток времени;
- рассчитывать показатели работы;
- и др.

### **Выводы по первому разделу**

Выполнен анализ источников по теме работы. Рассмотрена классификация и описаны сущности и виды процессов в сфере оказания стоматологических услуг. Приведено обоснование необходимости и цели использования программного обеспечения для автоматизации работы стоматологической клиники. Проведен сравнительный анализ аналогичных решений. Выявлены тенденции развития информационных систем в стоматологии, принято решение о проектировании приложения.

## 2 ПРОЕКТНАЯ ЧАСТЬ

### 2.1 Выбор инструментария решения задачи

Для проектирования и реализации данного проекта были изучены различные технологии: ASP.NET, Bootstrap, JQuery так же были получены знания по взаимодействию базы данных, а именно Microsoft SQL Server 2018.

ASP.NET – это платформа для создания Web– приложений и Web–сервисов, работающих под управлением IIS. Сегодня существуют другие технологии, позволяющие создавать Web– приложения. К ним относятся прежде всего, очень популярные сегодня языки PHP и PERL, более старая и менее популярная технология CGI и т. д. Однако ASP.NET отличается от них высокой степенью интеграции с серверными продуктами, а также с инструментами Microsoft для разработки доступа к данным и обеспечения безопасности. ASP.NET 5 представляет собой новую версию революционной технологии ASP.NET от Microsoft. Она является основным стандартом для создания динамических веб– страниц на платформе Windows. А в качестве языка реализации был выбран C#. Данный язык является последней разработкой из широко популярных языков программирования. Он вобрал в себя весь существующий опыт и лучшие стороны имеющихся языков программирования. У его разработчиков была возможность (которой они и воспользовались), оставить в прошлом все негативные особенности (как правило, это обратная совместимость), любого из предыдущих языков. C# включает много полезных свойств– простота, ООП, типовая защищенность, «сборка мусора», поддержка совместимости версий. Данные особенности дают возможность быстро и легко разрабатывать приложения. Язык C# специально создавался для работы в .NET платформе. Сама архитектура .NET продиктовала ему (как и большинству другим языкам, на которых можно писать под .NET) объектно–ориентированную направленность. Как результат мы имеем действительно

простой, удобный и современный язык, по мощности не уступающий C++, но существенно повышающий продуктивность разработок.

В качестве среды разработки использовалась программа VisualStudioCommunity2015. VisualStudio даёт возможность создавать программы для разных ОС, мобильных платформ, веб-сайты и приложения. Отличительным свойством встроенного редактора кода является удобная навигация, дружелюбный интерфейс. Порядок расстановки окон инструментов можно запомнить, создав макет экрана. Присутствует подсветка разметки и быстрый переход с помощью полос прокрутки. У всех приложений есть доступ к отладке, даже если в них используется не один язык программирования. Есть возможность остановки приложения и получения значений переменных на любом этапе выполнения программы. Для тестирования мобильных программ доступен эмулятор Android. Для написания игр есть поддержка движка Unity. Автоматический анализ и рефакторинг кода осуществляется с помощью компилятора Roslyn.

Преимущества Visual Studio:

- поддержка большого числа языков программирования;
- доступна бесплатная версия Community;
- Запись работы в режиме отладки приложения;
- доступна система для скачивания расширений;

Недостатки:

- большая загруженность оперативной памяти;
- файлы регистрации занимают большой объём памяти на жёстком диске.

Для хранения и работы с данными приложения задействована реляционная база данных Microsoft SQL Server 2018.

Microsoft SQL Server – система управления реляционными базами данных, разработанная корпорацией Microsoft. Основной используемый язык запросов – Transact-SQL, создан совместно Microsoft и Sybase. Transact-SQL

является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия; конкурирует с другими СУБД в этом сегменте рынка.

### **2.3 Характеристики деятельности стоматологической клиники**

Специалист управляющего отдела занимается вводом данных о клиентах, контрагентах в систему, выгрузкой и анализом отчетностей, отражающих картину деятельности клиники.

В обязанности специалистов управляющего отдела входят следующие функции [16]:

- обеспечение и организация лечебно-профилактической, хозяйственно-административной и финансовой деятельности учреждения;
- анализ показателей работы клиники;
- контроль смежных служб;
- организация качественного сервиса по работе с пациентами;
- мотивация сотрудников;
- разработка маркетинговой политики компании на основе анализа потребительского спроса;
- предоставление интересов организации в органах государственного управления.

Медицинский персонал осуществляет основной бизнес-процесс – оказание стоматологических услуг. Подразделение состоит из стоматологов разных профилей и их ассистентов. В зависимости от профиля врача его обязанности и обязанности ассистента могут меняться, но общими будут:

- проведение амбулаторного приема по графику;
- обеспечивать квалифицированное и своевременное обследование и лечение больных с заболеваниями зубов и полости рта;

- направлять больных, при наличии показаний, на дополнительные виды исследования;

- соблюдать в своей работе принципы деонтологии;

- осуществлять контроль и руководство работой младшего медицинского персонала (ассистентов) стоматологического кабинета;

- вести медицинские карты стоматологических больных.

В обязанности младшего медицинского персонала входят следующие функции [15]:

- подготовка кабинета и материалов, необходимых для приема пациентов;

- соблюдение правил санитарно-гигиенического и противоэпидемического режима, асептики;

- осуществление контроля чистоты и порядка в кабинете;

- проведение дезинфекции, предстерилизационной очистки и стерилизации инструментов;

- выполнение распоряжений врача во время приема пациентов;

- помощь врачу во время приема пациента по заполнению необходимой

- медицинской документации;

- осуществление контроля наличия и правильности заполнения всех документационных форм (наличие необходимых подписей, заполненной пациентом документации и т.п.);

Контекстная диаграмма деятельности стоматологической клиники отражена на рисунке 2.



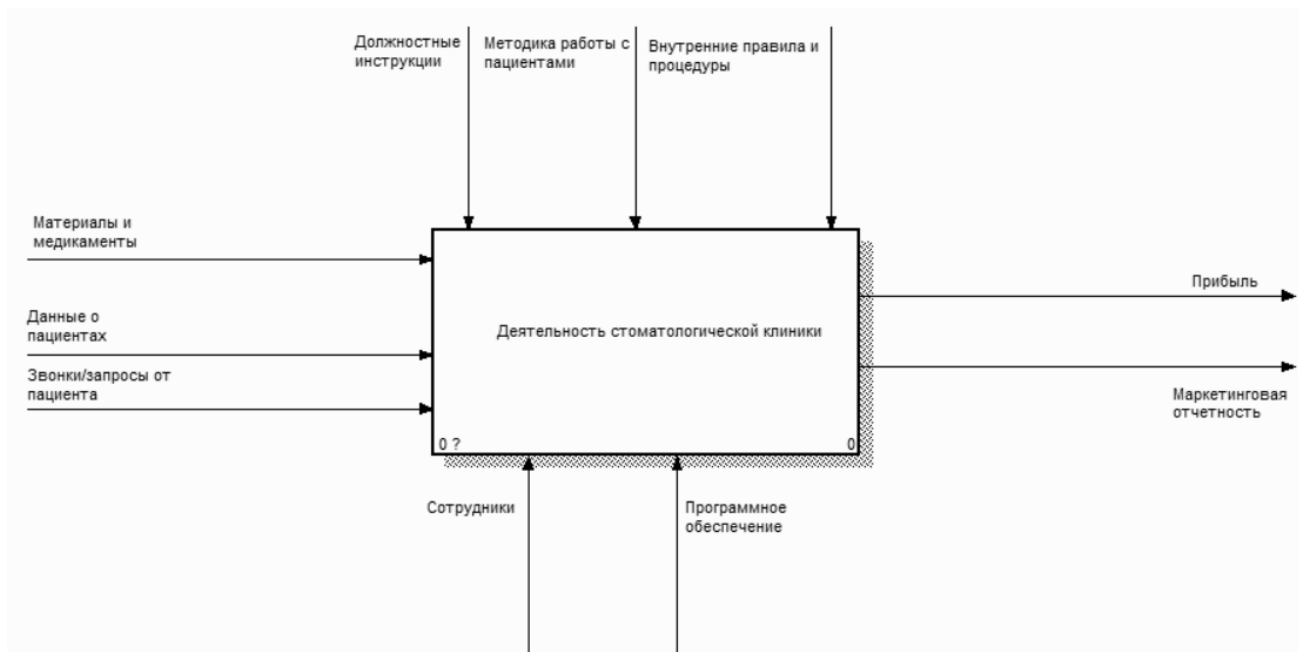


Рисунок 2– Контекстная диаграмма деятельности стоматологической клиники.

На контекстной диаграмме можно наблюдать потоки входа материалы, данные о пациентах, звонки/запросы от пациентов). На потоках выхода находится то, что выдает клиника (прибыль, маркетинговая отчетность) [21]. В потоки управления входят должностные инструкции, методика работы с пациентами, внутренние правила и процедуры, а основные ресурсы – это персонал и оборудование. После формирования контекстной диаграммы производится ее декомпозиция [22]. Декомпозиция – это разделение сложного объекта на его составные части. С помощью диаграммы декомпозиции первого уровня можно указать, из каких более мелких работ состоит деятельность клиники. Деятельность стоматологической клиники отражена на рисунке 3.

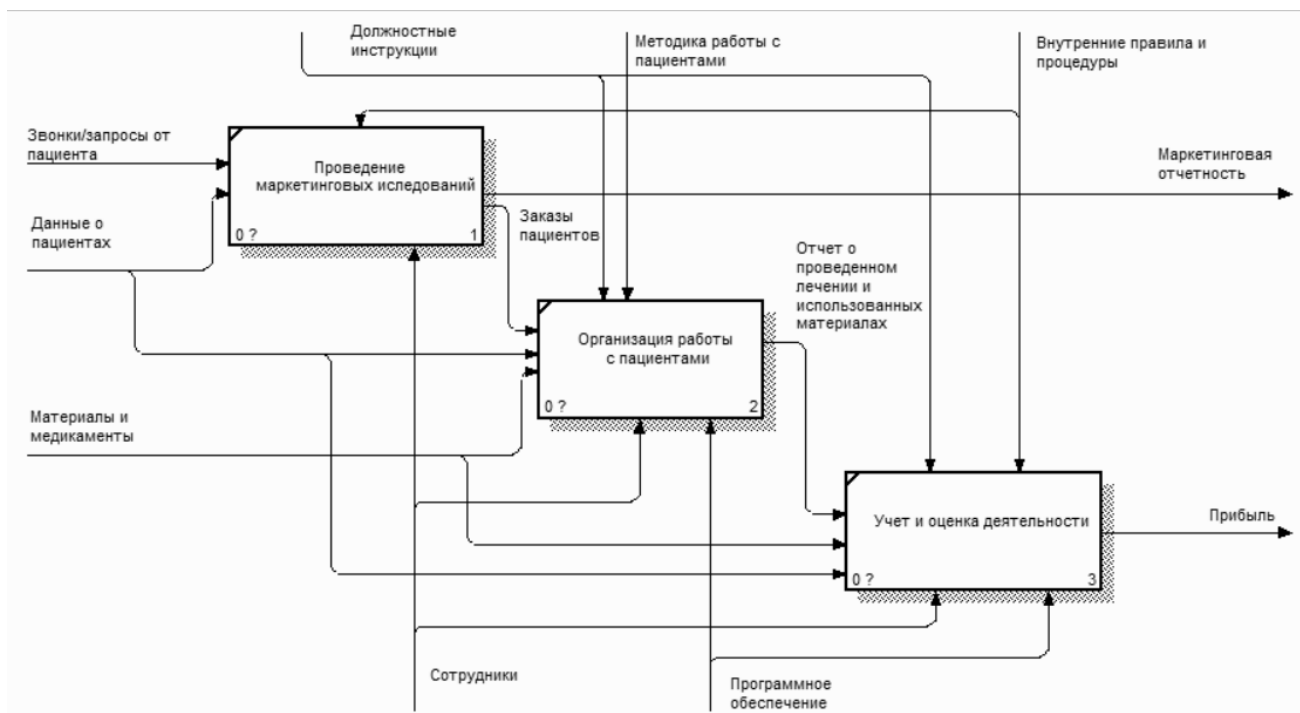


Рисунок 3– Диаграмма детализации деятельности клиники

## 2.4 Обоснование проектных решений

Основными причинами разработки и внедрения информационной системы в организации являются [25]:

- необходимость в получении более рациональных вариантов решения управленческих задач за счет внедрения математических методов и интеллектуальных систем и т.д.;
- повышение оперативности работы сотрудников организации;
- необходимость в замене бумажных носителей данных на магнитные диски, что приводит к более рациональной организации ИС и снижению объёмов документов на бумаге.

Программный продукт предназначен для автоматизации деятельности стоматологической клиники. Основные цели, которые должны быть достигнуты после внедрения информационной системы:

- повышение качества предоставления стоматологических услуг;

- планирование работы персонала, прием пациентов, закупки стоматологических и расходных материалов;
- повышение эффективности взаимодействия различных подразделений и сотрудников стоматологической клиники;
- унификация и систематизации данных о пациентах, персонале и оборудовании клиники, проведенных работах, использованных материалах, расчетах с клиентами;
- анализа и мониторинга деятельности клиники ее руководством и менеджментом для осуществления дальнейшего перспективного планирования и повышения эффективности бизнеса.

Установленные цели достигаются благодаря ряду следующих мероприятий:

- ведение единого учета клиентов (пациентов), единая форма ведения медицинской карты;
- реализация оптимизированного отчета, для быстрого получения информации для анализа;
- интеграция и взаимодействие различных информационных систем учета.

Основные требования заказчика к разработке:

- хранение и ведение информации о пациентах клиники, их медицинских картах;
- формирование отчетов и создание новых, удовлетворяющих требованиям организации;
- формирование и печать договоров, квитанций на оплату по требованию клиента и перед приемом;
- ведения учета материалов и лекарственных препаратов;
- ведение записи пациентов на приём.

Диаграмма потоков данных моделирует систему, как набор действий, и состоит из объектов, собирающих и хранящих информацию, и объектов, моделирующих взаимодействие с частями системы [23].

Контекстная диаграмма отражена на рисунке 4.

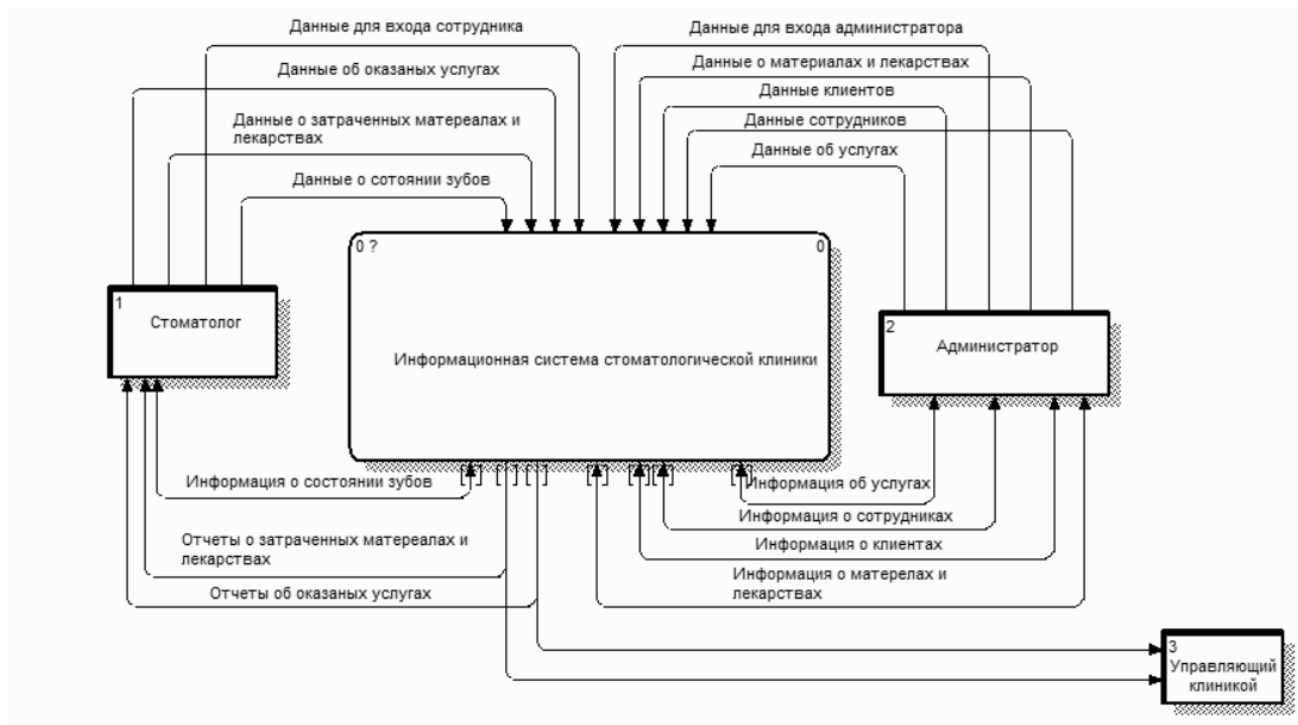


Рисунок 4– Контекстная диаграмма потоков данных

Структура информационной системы отражена на рисунке 5.

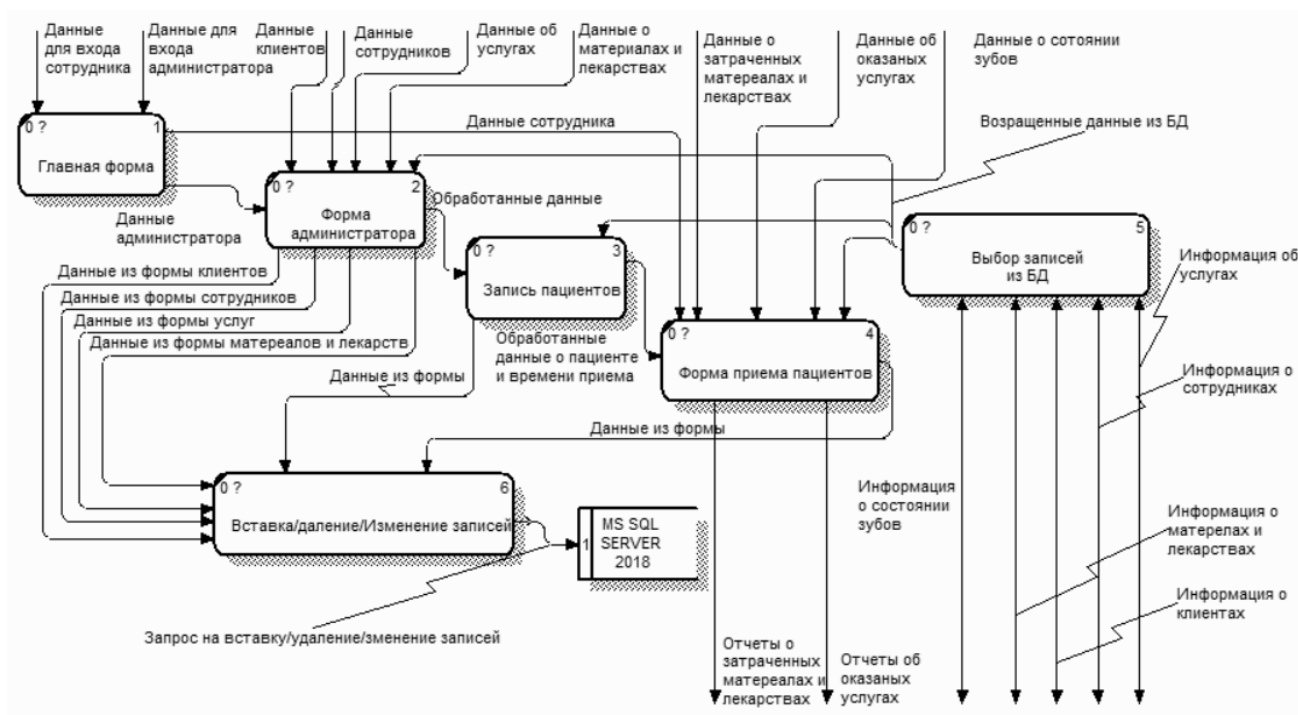


Рисунок 5– Диаграмма потоков данных

Создание системы автоматизации деятельности стоматологии позволит обеспечить:

- исключение дублирования данных о пациентах;
- сокращение времени на обработку и ввод данных;
- повышение качества предоставляемых услуг;
- получение данных о состоянии клиники.

Система должна обладать высокой скоростью обработки данных, гибкостью, надёжностью. Простота эксплуатации и поддержки системы, в будущем, должны сократить временные затраты на внедрение.

### 2.3 Структура базы данных

Для реализации проекта была выбрана реляционная база данных Microsoft SQL Server 2019. Разработка структуры базы данных включает в себя следующие шаги:

- определение состава и структуры базовой (универсальной) таблицы;

- определение логических связей, первичных ключей и функциональных зависимостей между атрибутами (полями) базовой таблицы;
- нормализация (разделение) базовой таблицы;
- определение состава, структуры таблиц;
- выбор первичных ключей (ключевых полей),
- нормализация таблиц, определение типов данных;
- определение типа связей между таблицами.

Реализация схемы базы данных, для организации бизнес – процессов фирмы, представлена на рисунке 6.

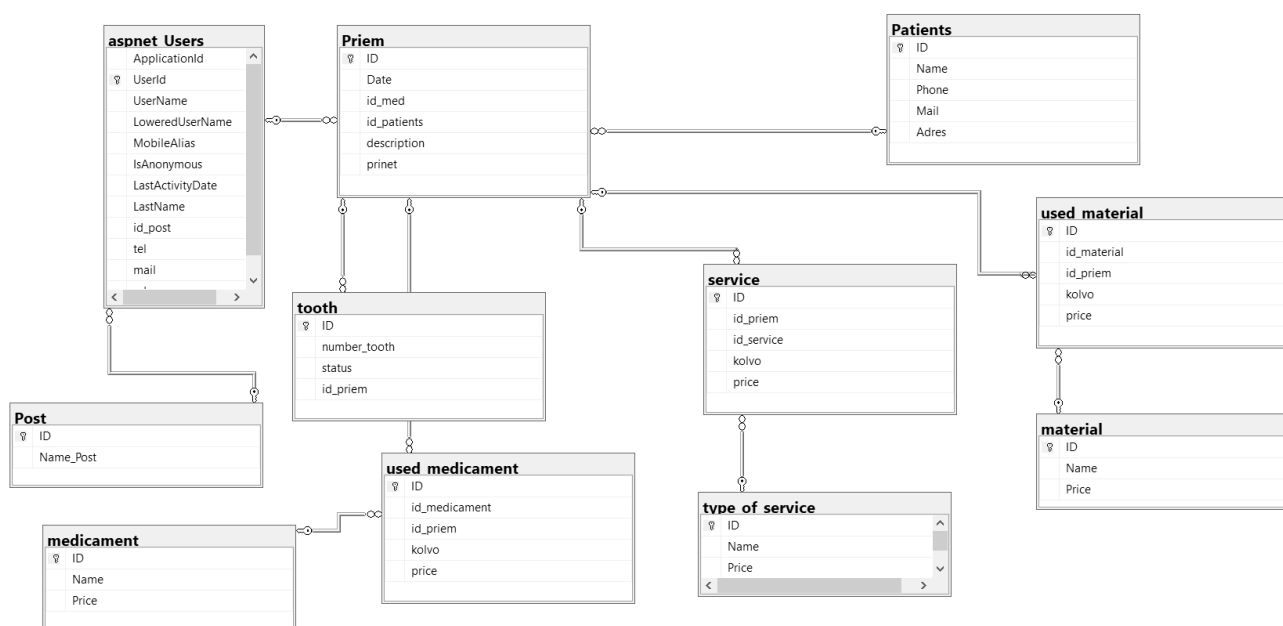


Рисунок 6 – Схема базы данных

Таблица «Post» содержит должности сотрудников стоматологической клиники:

- Врач стоматолог – ортопед;
- стоматолог – терапевт;
- Врач стоматолог общей практики;
- ассистент стоматолога.

Список её полей приведён в таблице 2.

Таблица 2 – Поля таблицы «Post»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
Name_Post	Наименование	Текстовое

В таблице «tooth» содержится информация о номере зуба и его состоянии. Список её полей приведён в таблице 3.

Таблица 3 – Поля таблицы «tooth»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
number_tooth	Номер зуба	Целое
status	Состояние зуба	Текстовое
Id_priem	Код приема	Целое

Таблица «type\_of\_service» хранит информацию о наименовании и стоимости доступных услуг стоматологической клиники. Список её полей приведён в таблице 4.

Таблица 4 – Поля таблицы «type\_of\_service»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
Name	Наименование услуги	Текстовое
Price	Стоимость услуги	Вещественное

Таблица «service» содержит информацию об оказанных услугах для пациента и их стоимости. Список её полей приведён в таблице 5.

Таблица 5 – Поля таблицы «service»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
id_service	Код услуги	Целое
id_priem	Код приема	Целое
kolvo	Количество услуг	Целое
price	Стоимость услуги	Вещественное

Таблица «material» хранит информацию о наименовании и стоимости материалов. Список её полей приведён в таблице 6.

Таблица6 – Полятаблицы «material»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
Name	Наименование материала	Текстовое
Price	Стоимость материала	Вещественное

Таблица «used\_material» содержит информацию об использованных материалах для пациента и их стоимости. Список её полей приведён в таблице 7.

Таблица7 – Полятаблицы «used\_material»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
id_material	Код материала	Целое
id_priem	Код приема	Целое
kolvo	Количество материалов	Целое
price	Стоимость материалов	Вещественное

Таблица «Patients» хранит информацию о пациентах. Список её полей приведён в таблице 8.

Таблица8 – Полятаблицы «Patients»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
Name	ФИО	Текстовое
Phone	Телефон	Текстовое
Mail	Email	Текстовое
Adres	Адрес	Текстовое

В таблице «Priem» регистрируется информация о дате приема пациента, лечащем враче, описание причины посещения и статусе(принят, не принят) приемаврачом. Список её полей приведён в таблице 9.



Таблица9 – Полятаблицы «Priem»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
Date	Дата приема	Дата
id_med	Код сотрудника	Текстовое
id_patients	Код пациента	Текстовое
description	Описание приема	Текстовое
prinet	Статус	Чекбокс

Таблица «medicament» хранитинформацию о наименовании и стоимости лекарств. Список её полей приведён в таблице 10.

Таблица10 – Полятаблицы «medicament»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
Name	Наименование лекарства	Текстовое
Price	Стоимость лекарства	Вещественное

Таблица «used\_medicament» содержитинформацию об использованных лекарствах для пациента и их стоимости. Список её полей приведён в таблице 11.

Таблица11 – Полятаблицы «used\_medicament»

Наименование поля	Описание поля	Тип поля
ID	Уникальный код	Идентификатор
id_medicament	Код лекарства	Целое
id_priem	Код приема	Целое
kolvo	Количество лекарств	Целое
price	Стоимость лекарства	Вещественное

Таблица «aspnet\_Users» содержитинформацию о сотрудниках. Список её полей приведён в таблице 12.

Таблица12 – Полятаблицы «aspnet\_Users»

Наименование поля	Описание поля	Тип поля
UserId	Уникальный код	Идентификатор
LastName	ФИО	Текстовое
id_post	Код должности	Целое
tel	телефон	Текстовое
mail	Email	Текстовое
adres	адрес	Текстовое

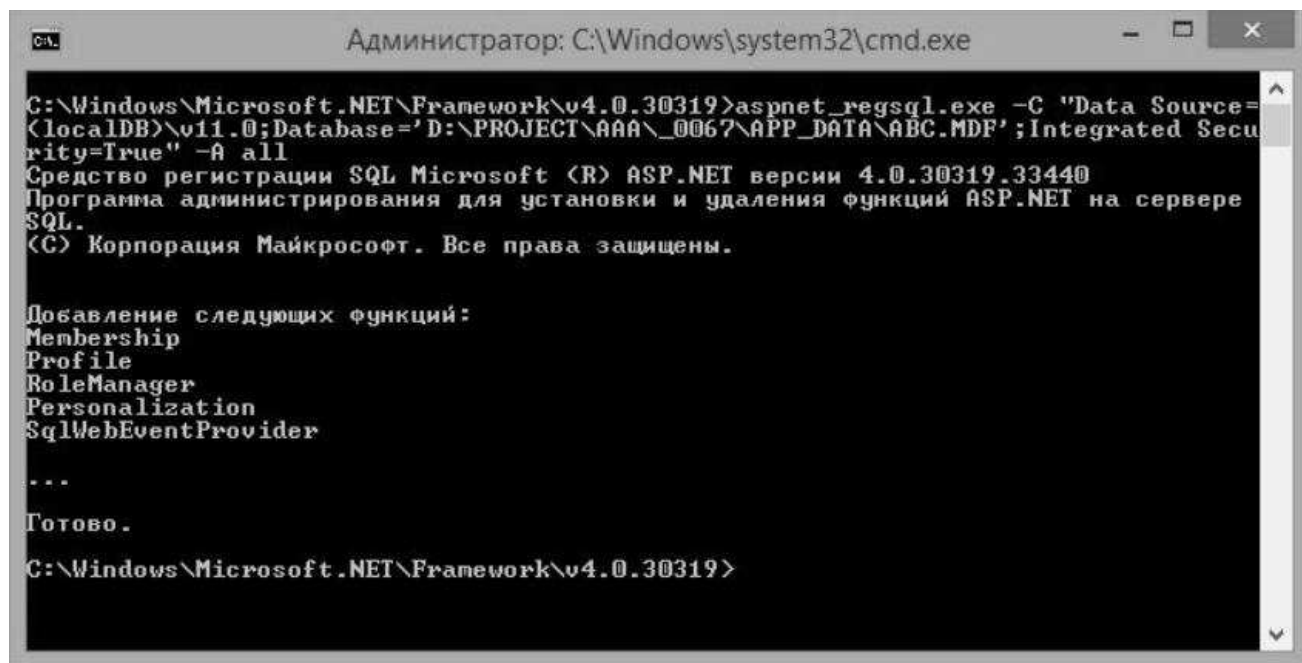
## 2.4 Создание схемы членства в SQL Server

До ASP.NET 2,0 разработчики были в процессе реализации задач, связанных с учетной записью пользователя. Команда ASP.NET решила эту задачу и представила инфраструктуру членства с ASP.NET 2,0. Платформа членства— это набор классов в платформе .NET Framework, предоставляющих программный интерфейс для выполнения основных задач, связанных с учетной записью пользователя. Эта платформа реализована на основе модели поставщика, которая позволяет разработчикам подключать пользовательскую реализацию к стандартизированному API.

Как обсуждалось в учебнике основы безопасности и поддержка ASP.NET, платформа .NET Framework поставляется с двумя встроенными поставщиками членства: `ActiveDirectoryMembershipProvider` и `SqlMembershipProvider`. Как следует из названия, `SqlMembershipProvider` компонент использует базу данных Microsoft SQL Server в качестве хранилища пользователей. Чтобы использовать этот поставщик в приложении, необходимо сообщить поставщику, какую базу данных использовать в качестве хранилища. База данных хранилища пользователей для `SqlMembershipProvider` должна иметь определенные таблицы, представления и хранимые процедуры базы данных. Необходимо добавить эту схему в выбранную базу данных.

Для `SqlMembershipProvider` необходимо установить определенный набор таблиц, представлений и хранимых процедур в базу данных хранилища пользователей. Эти необходимые объекты базы данных можно добавить с помощью `aspnet_regsql.exe` утилиты. Этот файл находится в `%WINDIR%\Microsoft.Net\Framework\v2.0.50727\aspnet_regsql.exe`. Средство `aspnet_regsql.exe` используется для добавления или удаления служб приложений ASP.NET в указанной базе данных SQL Server. Службы приложений ASP.NET охватывают схемы для `SqlMembershipProvider` и `SqlRoleProvider`.

Чтобы запустить aspnet\_regsql.exe утилиту, перейдите в меню «Пуск» и выберите команду «выполнить». Введите %WINDIR%\Microsoft.Net\Framework\k\v2.0.50727\aspnet\_regsql.exe в текстовое поле и нажмите кнопку ОК– рисунок 7.



```
Администратор: C:\Windows\system32\cmd.exe
C:\Windows\Microsoft.NET\Framework\v4.0.30319>aspnet_regsql.exe -C "Data Source=
(localDB)\v11.0;Database='D:\PROJECT\AAA\0067\APP_DATA\ABC.MDF';Integrated Secu
rity=True" -A all
Средство регистрации SQL Microsoft (R) ASP.NET версии 4.0.30319.33440
Программа администрирования для установки и удаления функций ASP.NET на сервере
SQL.
(C) Корпорация Майкрософт. Все права защищены.

Добавление следующих функций:
Membership
Profile
RoleManager
Personalization
SqlWebEventProvider
...
Готово.
C:\Windows\Microsoft.NET\Framework\v4.0.30319>
```

Рисунок 7– Запуск aspnet\_regsql.exe утилиты с аргументами

Запуск aspnet\_regsql.exe средства без аргументов командной строки запускает графический пользовательский интерфейс мастера установки SQL Server ASP.NET. Мастер упрощает добавление или удаление служб приложений ASP.NET в указанной базе данных. На первом экране мастера, показанном на рисунке 8, описывается назначение этого средства.

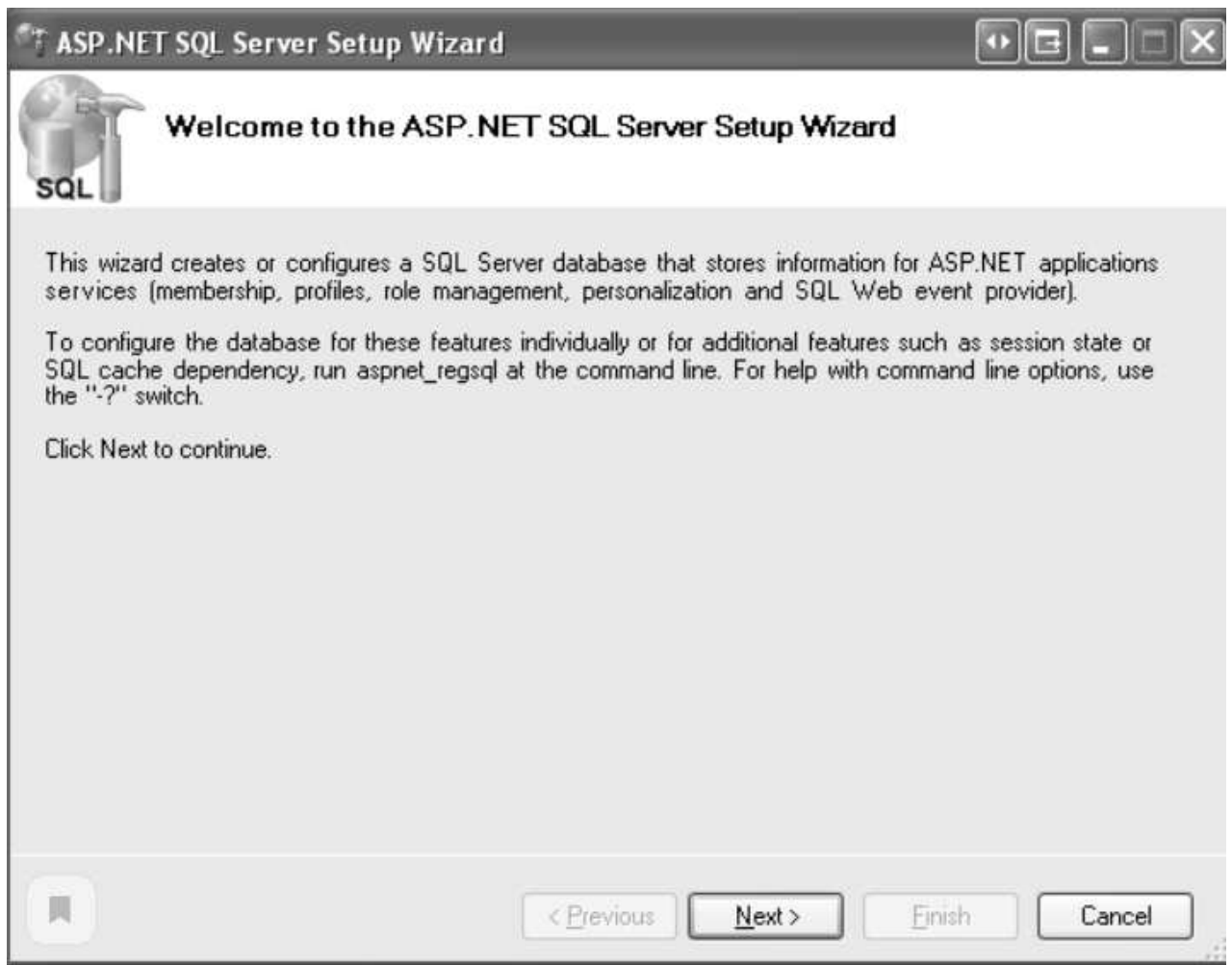


Рисунок8 – Использование мастера установки ASP.NET SQL Server для добавления схемы членства

На втором шаге мастера запрашивается, нужно ли добавить службы приложений или удалить их. Так как мы хотим добавить таблицы, представления и хранимые процедуры, необходимые для SqlMembershipProvider, выберите параметр настроить SQL Server для служб приложений. Позже, если вы хотите удалить эту схему из базы данных, перезапустите этот мастер, а вместо этого выберите параметр удалить сведения о службах приложений из существующей базы данных – рисунок 9.

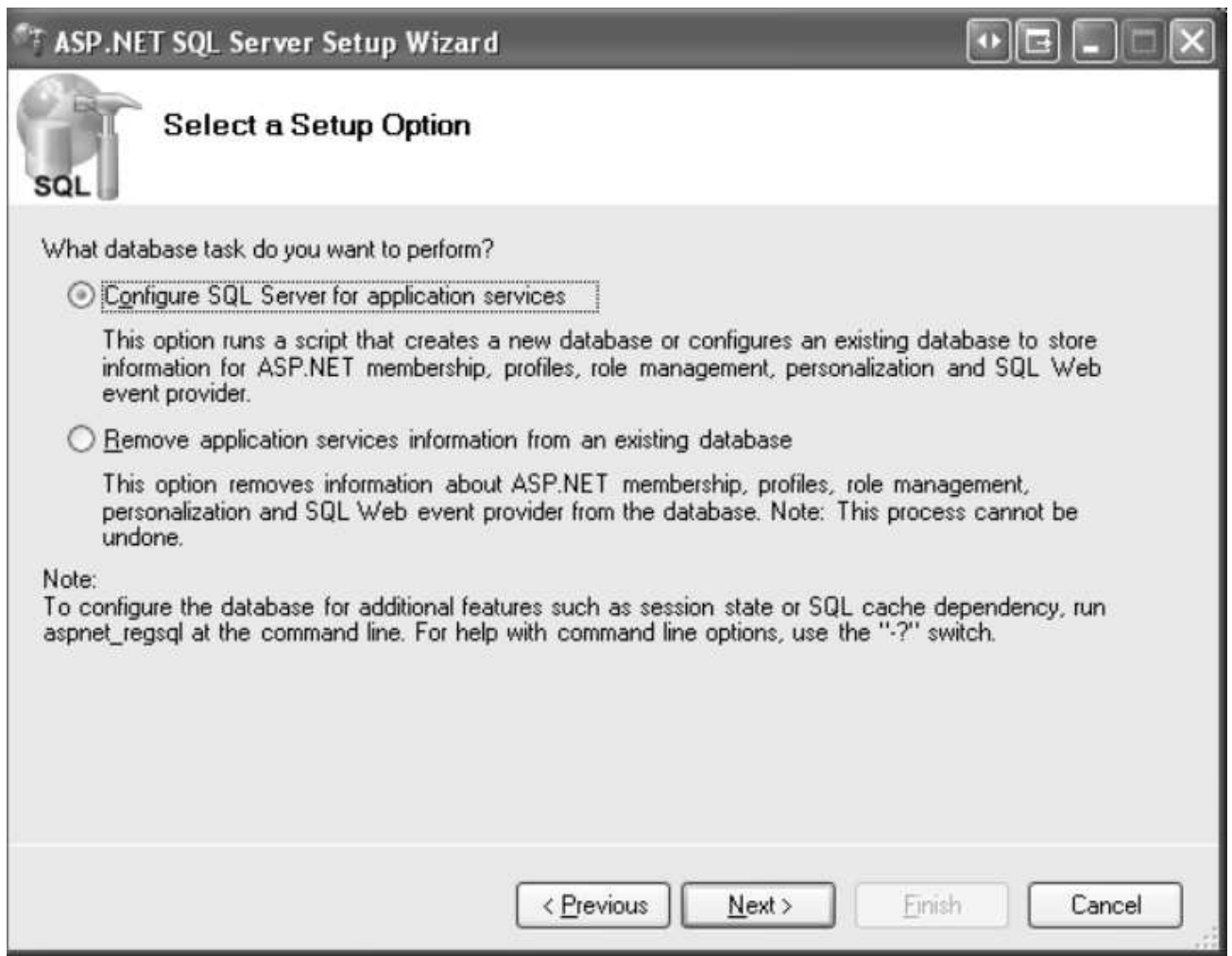


Рисунок 9 – Выбор параметра настройки SQL Server для службы приложений

На третьем шаге запрашиваются сведения о базе данных: имя сервера, сведения о проверке подлинности и имя базы данных – рисунок 10

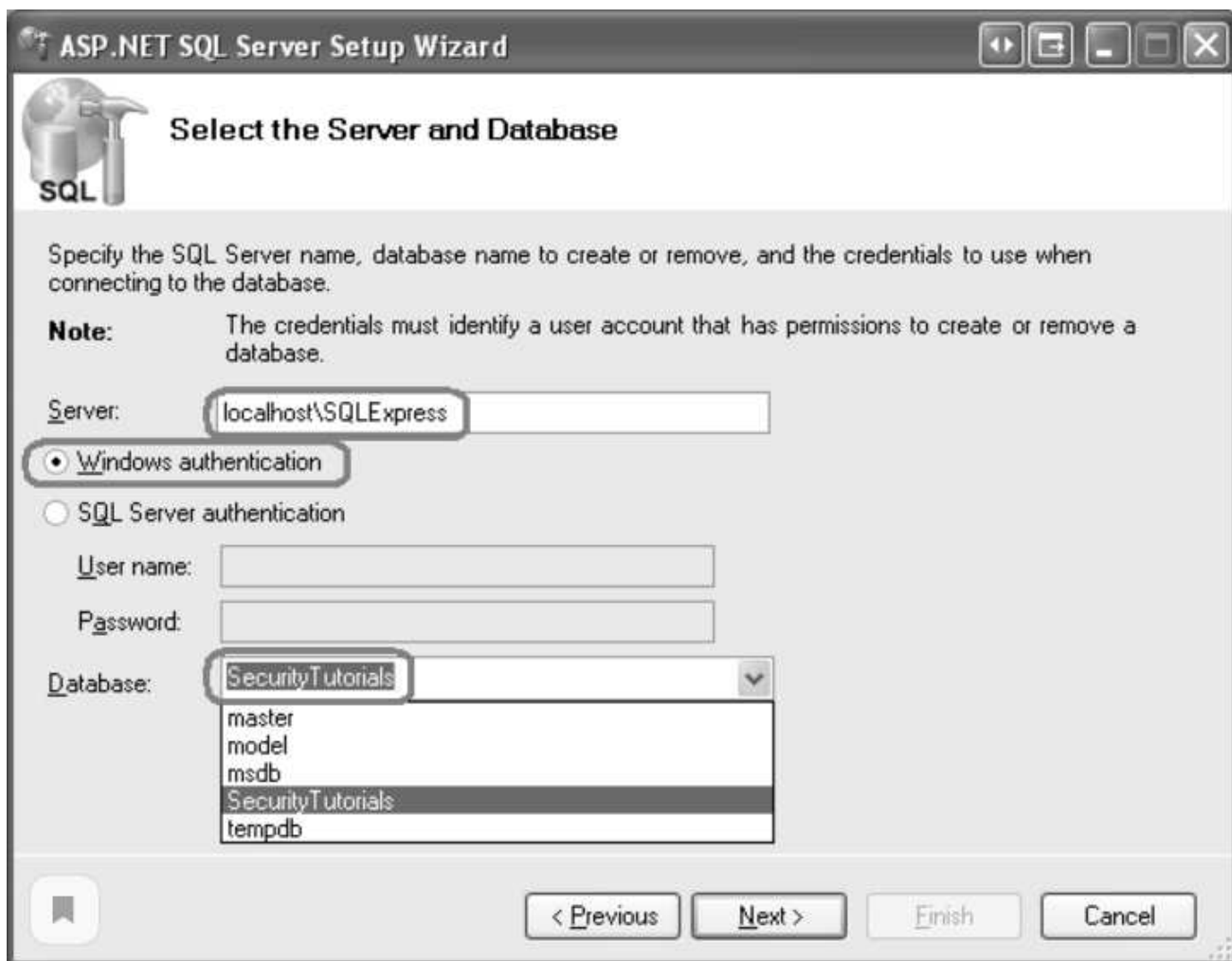


Рисунок 10 – Ввод сведений о базе данных

После ввода сведений о базе данных нажмите кнопку Далее. На последнем шаге выполняется сводка действий, которые будут выполнены. Нажмите кнопку Далее, чтобы установить службы приложений и завершить работу мастера.

После завершения работы мастера вернитесь в Visual Studio и перейдите к обозревателю базы данных. Разверните папку Таблицы. Вы увидите ряд таблиц, имена которых начинаются с префикса aspnet\_. Аналогично, в папках views и хранимые процедуры можно найти разнообразные представления и хранимые процедуры. Эти объекты базы данных составляют схему служб приложений – рисунок 11.



Рисунок 11– В базу данных добавлены разнообразные таблицы, представления и хранимые процедуры

На этом этапе мы создали объекты базы данных, необходимые для SqlMembershipProvider.

Структуры членства и ролей спроектированы таким образом, что один пользователь и хранилище ролей могут совместно использоваться разными приложениями. Приложение ASP.NET, использующее платформы членства или ролей, должно указывать, какую секцию приложения следует использовать. Вкратце, несколько веб-приложений могут использовать одни и те же хранилища пользователей и ролей. На рисунке 12 показаны хранилища пользователей и ролей, разделенные на три приложения: HRSite, CustomerSite и SalesSite. Каждый из этих трех веб-приложений имеет собственные уникальные пользователи и роли, но все они физически хранят свои учетные записи и сведения о ролях в одних и тех же таблицах базы данных.

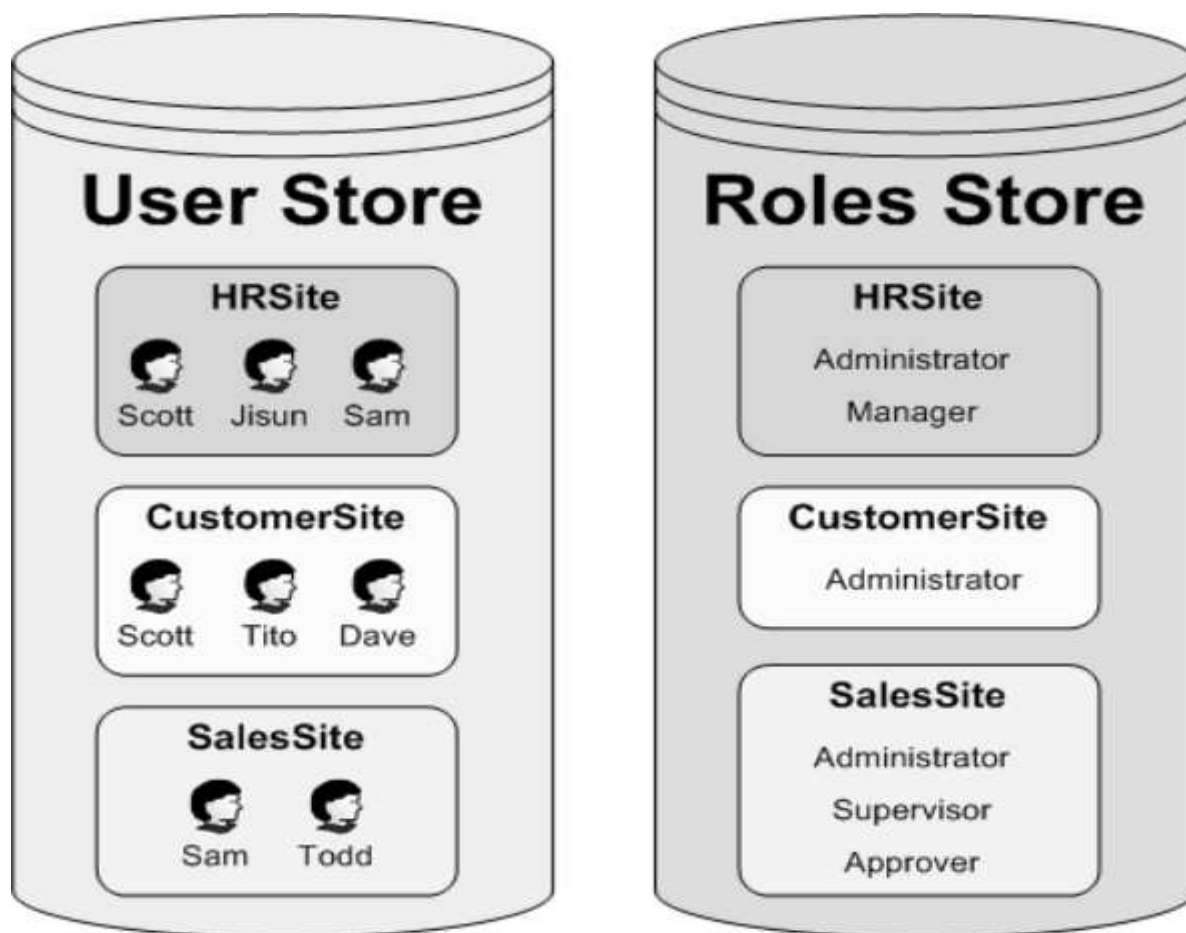


Рисунок 12– Учетные записи пользователей могут быть разделены между несколькими приложениями



Таблица `aspnet_Applications` определяет эти секции. Каждое приложение, использующее базу данных для хранения данных учетной записи пользователя, представлено строкой в этой таблице. Таблица состоит из четырех столбцов: `ApplicationId`, `ApplicationName`, `LoweredApplicationName` и `Description`. `ApplicationId` имеет тип `uniqueidentifier` и является первичным ключом таблицы; `ApplicationName` предоставляет уникальное понятное имя для каждого приложения.

Другие таблицы, относящиеся к членству и ролям, связываются с полем `ApplicationId` в `aspnet_Applications`. Например, таблица `aspnet_Users`, содержащая запись для каждой учетной записи пользователя, имеет поле `ApplicationId` внешнего ключа, также для таблицы `aspnet_Roles`. В поле `ApplicationId` в этих таблицах указывается секция приложения, к которой принадлежит учетная запись пользователя или роль.

Сведения об учетной записи пользователя размещены в двух таблицах: `aspnet_Users` и `aspnet_Membership`. Таблица `aspnet_Users` содержит поля, в которых хранятся сведения о наиболее ценных учетных записях пользователей. Ниже перечислены три наиболее соответствующих столбца:

- `UserId`;
- `UserName`;
- `ApplicationId`.

`UserId` является первичным ключом (и типом `uniqueidentifier`). `UserName` имеет тип `nvarchar(256)` и, вместе с паролем, создает учетные данные пользователя. Пароль пользователя хранится в `aspnet_Membership` таблице. `ApplicationId` связывает учетную запись пользователя с определенным приложением в `aspnet_Applications`. Существует `UNIQUE` ограничение для `UserName ApplicationId`. Это гарантирует, что каждое имя пользователя уникально в данном приложении, но также позволит использовать его `UserName` в нескольких проектах.

Таблица `aspnet_Membership` содержит дополнительные сведения об учетной записи пользователя, такие как пароль пользователя, адрес электронной почты, Дата и время последнего входа в систему и т. д. Между записями в таблицах `aspnet_Users` и `aspnet_Membership` есть соответствие один к одному. Эта связь обеспечивается полем `UserId` в `aspnet_Membership` таблице, которое служит первичным ключом таблицы. Как и в `aspnet_Users` таблице, `aspnet_Membership` включает `ApplicationId` поле, которое связывает эту информацию с определенной секцией приложения. Информация о пароле хранится в `aspnet_Membership` таблице. `SqlMembershipProvider` позволяет хранить пароли в базе данных с помощью одного из следующих трех методов:

**Clear** – пароль хранится в базе данных в виде обычного текста.

**Хэшированные** – пароли хэшируются с использованием одностороннего алгоритма хэширования и случайным образом сформированным расширяющим значением. Это хэшированное значение (вместе с `Salt`) хранится в базе данных.

**Зашифровано** – зашифрованная версия пароля хранится в базе данных.

Используемый способ хранения паролей зависит от параметров, `SqlMembershipProvider` указанных в `Web.config`.

Столбцы, отвечающие за хранение пароля `Password`, `PasswordFormat` и `PasswordSalt`. `PasswordFormat` поле типа `int`, значение которого указывает метод, используемый для хранения пароля: 0 для `Clear`; 1 – для хэширования; 2 – для шифрования. `PasswordSalt` присваивается случайным образом сформированная строка независимо от используемого метода хранения паролей. значение `PasswordSalt` используется только при вычислении хэша пароля. Наконец, `Password` столбец содержит фактические данные пароля, то есть пароль в виде обычного текста, хэш пароля или зашифрованный пароль.

В таблице 13 показано, как могут выглядеть эти три столбца для различных методов хранения при хранении пароля `MySecret!`.

Таблица 13– Пример хранения пароля mysecret!

Метод хранения	Значение	Формат
Clear	MySecret!	0
Хэшируется	2oXm6sZHWbTHFgjkGQsc2Ec9ZM =	1
С шифрованием	62RZgDvhxykkqsMchZ0Yly7HS6onhpaoCYaRxV8g0F4CW56OXUU 3e7Inza9j9BKp	2

## 2.5Интерфейс программы

Главная веб – форма информационной системы показана на рисунке 13.

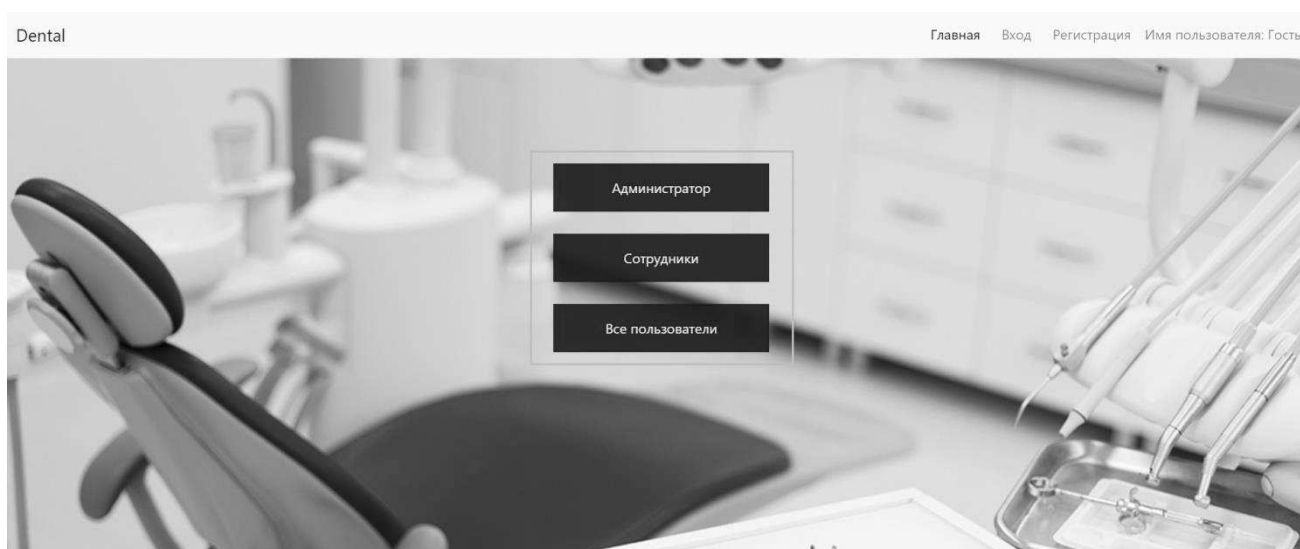


Рисунок 13 – Главная веб – форма

На веб – форме реализован доступ для разных категорий пользователей:

- Администратор;
- Сотрудники;
- Не зарегистрированные пользователи.

Страницы для входа и регистрации показаны на рисунке 14 и рисунке 15.



Рисунок 14 – Веб – форма для входа

Фрагмент кода для входа в систему описан в приложении А.

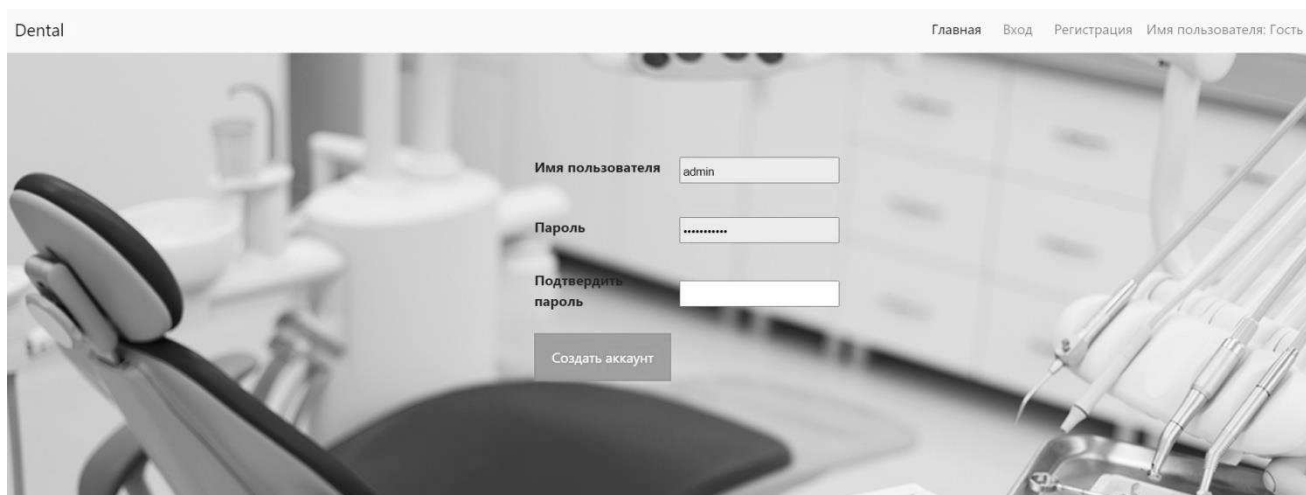


Рисунок 15 – Веб – форма для регистрации

Для организации работы с учетными записями пользователей используется класс Membership, пространство имен: System.Web.Security.

Класс Membership используется в приложениях ASP.NET для проверки учетных данных пользователя и управления параметрами пользователя, такими как пароли и адреса электронной почты. Класс можно использовать самостоятельно или в сочетании с FormsAuthentication, чтобы создать полную систему для проверки подлинности пользователей веб-приложения или сайта.

Элемент Login управления инкапсулирует класс, чтобы предоставить удобный механизм проверки пользователей.

КлассMembership предоставляет средства для:

- Создание новых пользователей;
- Хранение сведений о членстве (имена пользователей, пароли, адреса электронной почты и вспомогательные данные) в Microsoft SQL Server или в альтернативном хранилище данных;
- Проверка подлинности пользователей, посещающих ваш сайт. Вы можете выполнять проверку подлинности пользователей программно или с помощью Login элемента управления, чтобы создать полную систему проверки подлинности, требующую практически отсутствия кода;
- Управление паролями, включая создание, изменение, извлечение и сброс параметров, и т.д. При необходимости можно настроить членство в ASP.NET для запроса пароля и ответа для проверки подлинности при сбросе пароля или запросов на извлечение для пользователей, которые забыли пароль.

Главная веб – форма администратора представлена на рисунке 16.

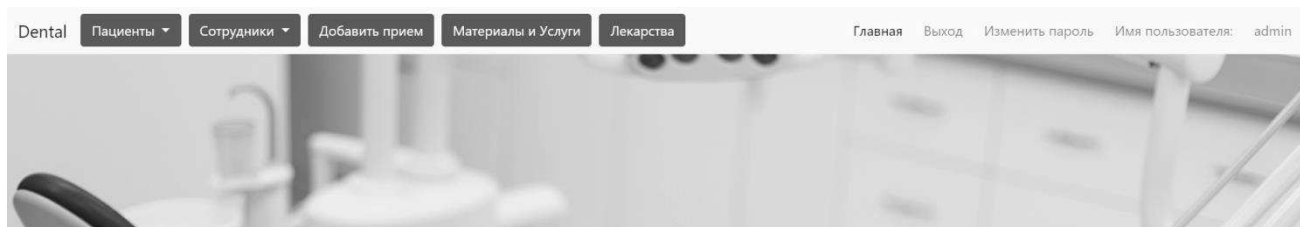


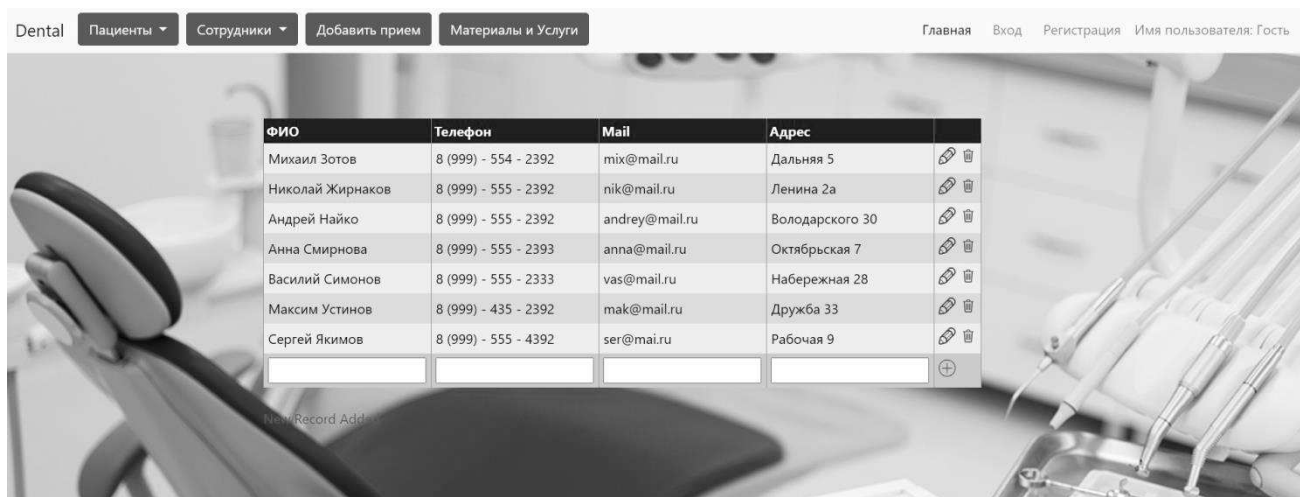
Рисунок 16 – Главная веб – форма администратора

На форме реализовано меню для организации работы со справочниками:

- пациенты;
- сотрудники;
- материалы;
- услуги;
- лекарства.

Также реализована форма для записи пациентов на прием.

Веб – форма для регистрации сотрудников показана на рисунке 17.



ФИО	Телефон	Mail	Адрес
Михаил Зотов	8 (999) - 554 - 2392	mix@mail.ru	Дальняя 5
Николай Жирнаков	8 (999) - 555 - 2392	nik@mail.ru	Ленина 2а
Андрей Найко	8 (999) - 555 - 2392	andrey@mail.ru	Володарского 30
Анна Смирнова	8 (999) - 555 - 2393	anna@mail.ru	Октябрьская 7
Василий Симонов	8 (999) - 555 - 2333	vas@mail.ru	Набережная 28
Максим Устинов	8 (999) - 435 - 2392	mak@mail.ru	Дружба 33
Сергей Якимов	8 (999) - 555 - 4392	ser@mai.ru	Рабочая 9

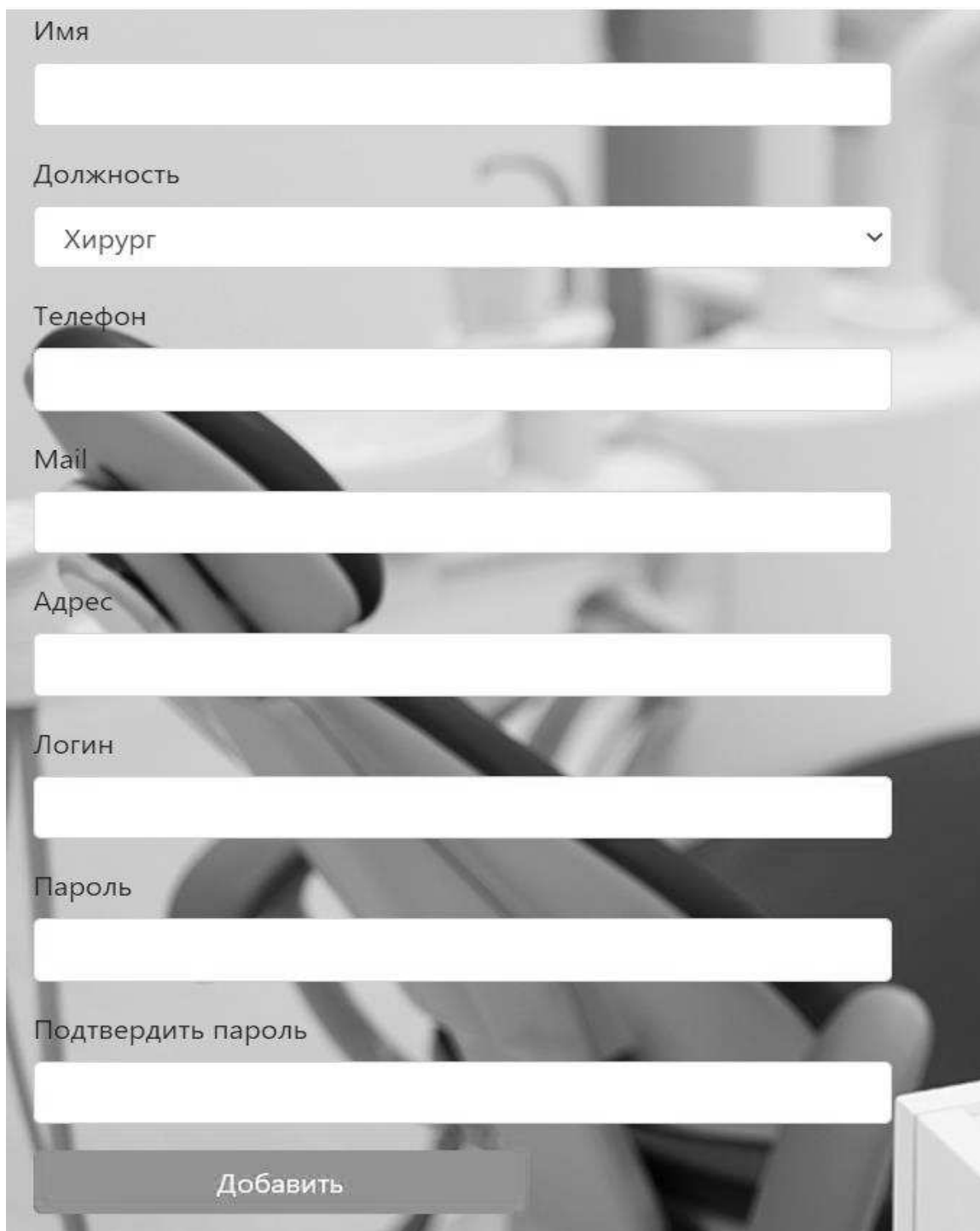
Рисунок 17 – Веб – форма пациенты

Форма содержит основную информацию о клиенте:

- ФИО;
- телефон;
- email;
- адрес.

Реализация выгрузки данных в таблицу осуществляется с помощью функции `PopulateGridView()`, код которой описан в приложении Б.

Страница для регистрации сотрудников показана на рисунке 18.



Имя

Должность

Телефон

Mail

Адрес

Логин

Пароль

Подтвердить пароль

Добавить

Рисунок 18 – Страница для регистрации сотрудников

Страница содержит форму для регистрации сотрудников в системе.

В форму вносится информация о сотруднике:

ФИО;

Должность;

- Телефон;
- Email;
- Адрес;
- Логин;
- Пароль.

Веб – форма всех зарегистрированных сотрудников изображена на рисунке 19.

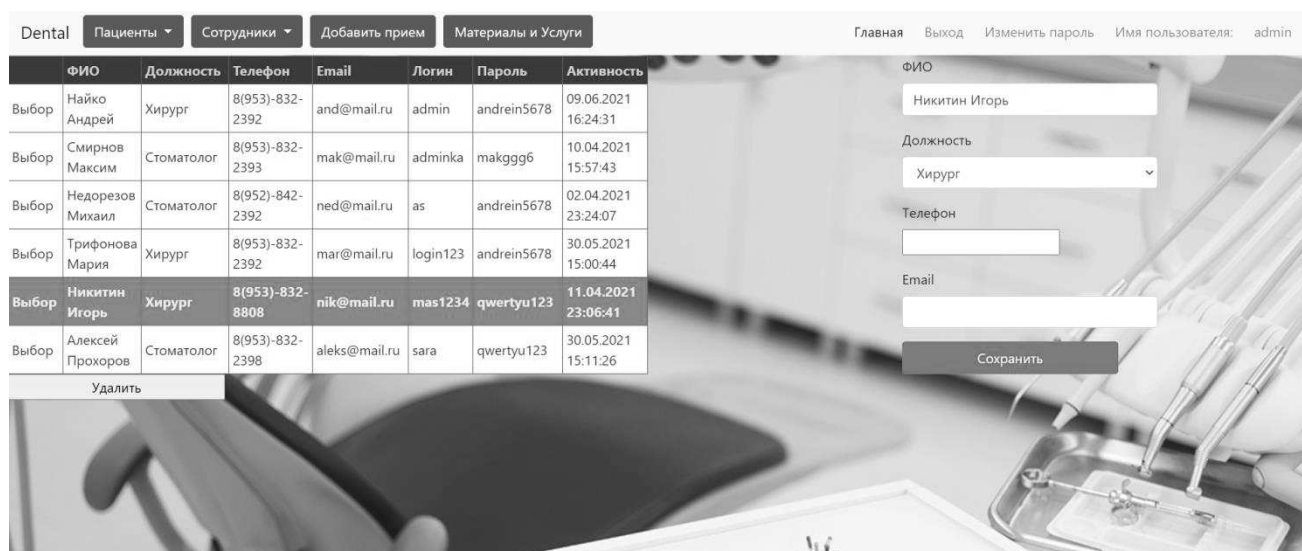


Рисунок 19 – Веб – форма всех зарегистрированных сотрудников

Форма содержит основную информацию о сотрудниках и дату последнего входа в систему. При выборе сотрудника, открывается, в правой части страницы, форма для редактирования.

Веб – форма справочников материалов и услуг показана на рисунке 20.



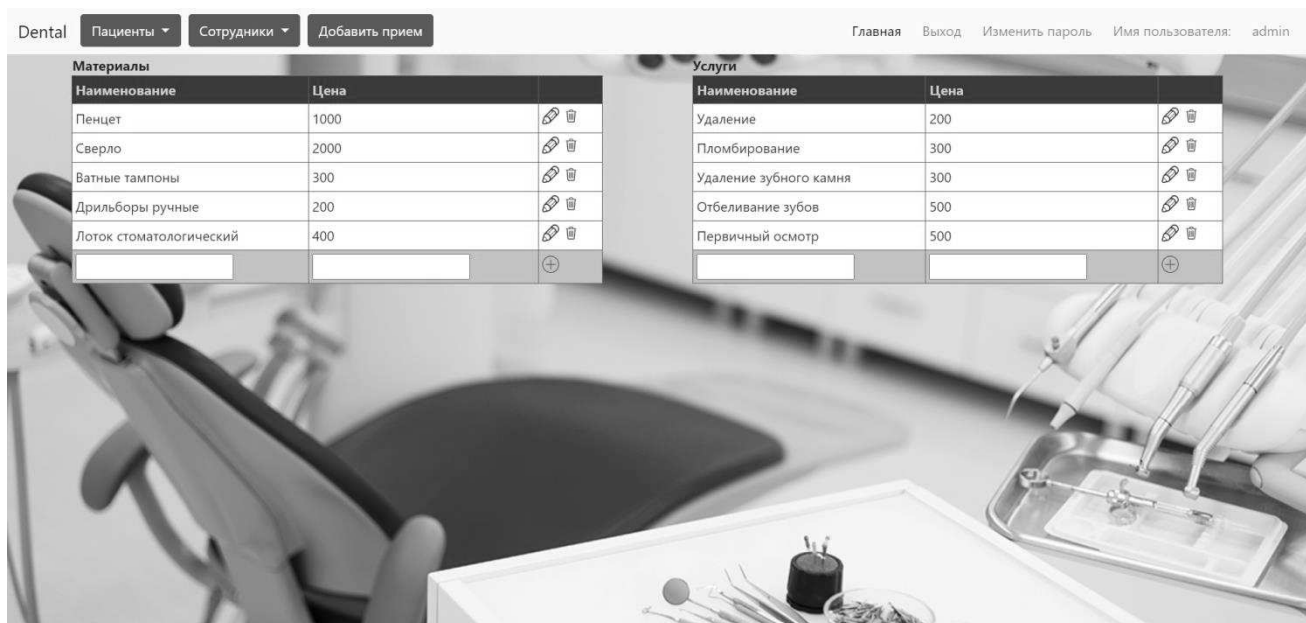


Рисунок 20 – Веб – форма справочников материалов и услуг

Форма содержит информацию о материалах и услугах, их наименовании и стоимости. Реализовано добавление, редактирование и удаление данных, фрагменты кода описаны в приложение В, приложение Г и приложение Д соответственно.

Страница записи пациентов на прием показана на рисунке 21.

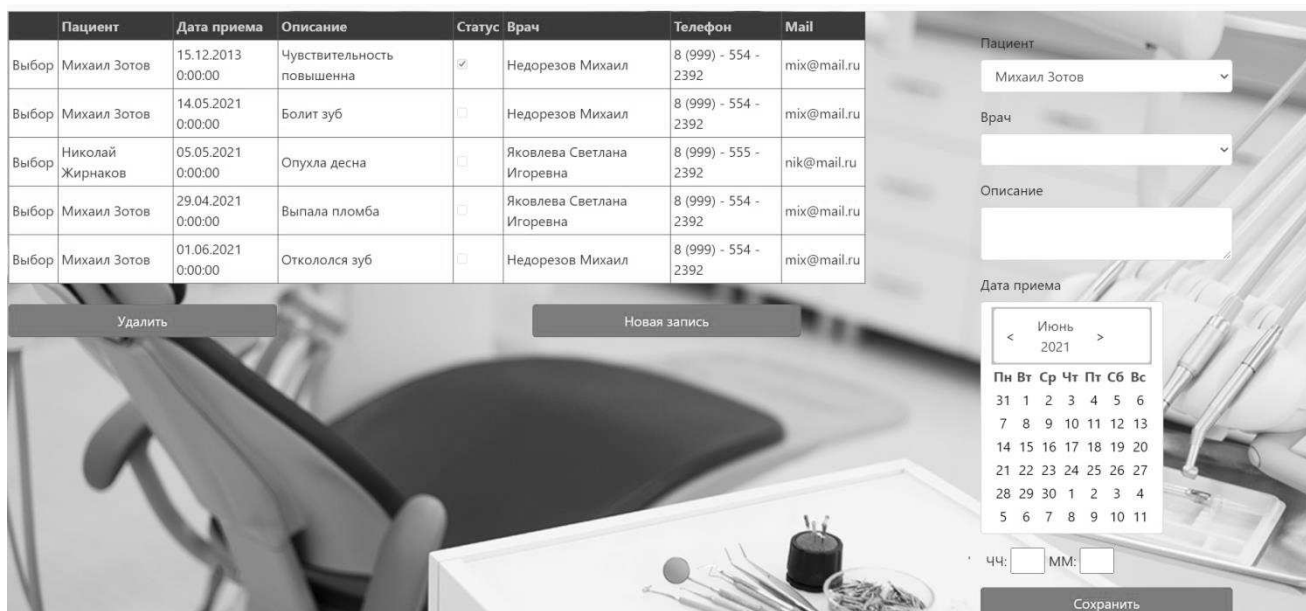


Рисунок 21 – Веб – форма записи пациентов на прием

На форме реализовано добавление, редактирование, удаление записей приемов пациентов. После добавления данных формируется отчет с информацией о пациенте, дате приема, статусе приема и лечащем враче.

Веб – форма для сотрудников показана на рисунке 22.

Пациент    Приемы на:

Выбор	Пациент	Дата приема	Описание	Статус	Врач	Телефон	Email
Выбор	Николай Жирнаков	05.05.2021 0:00:00	Опухла десна	<input type="checkbox"/>	Яковлева Светлана Игоревна	8 (999) - 555 - 2392	nik@mail.ru
Выбор	Михаил Зотов	29.04.2021 0:00:00	Выпала пломба	<input type="checkbox"/>	Яковлева Светлана Игоревна	8 (999) - 554 - 2392	mix@mail.ru

Статус приема:  
 Принят  
 Не принят

Материал  
 Пенцет  
 Количество

Выбор	Материал	Количество	Цена	Сумма
Выбор	Пенцет	3	100	300
Выбор	Пенцет	5	1000	5000
Выбор	Пенцет	2	1000	5000
Выбор	Сверло	2	2000	4000
Выбор	Пенцет	9	1000	9000
Выбор	Сверло	77	2000	154000
Выбор	Сверло	77	2000	154000
Выбор	Сверло	76	2000	152000
Выбор	Итого:			483300

Экспорт в Excel

Выбор	Лекарство	Количество	Цена	Сумма
Выбор	Мышьяк	3	300	900

Экспорт в Excel

Выбор	Услуга	Количество	Цена	Сумма
Выбор	Удаление	3	100	300
Выбор	Удаление	5	200	1000
Выбор	Удаление	7	200	1400
Выбор	Удаление	8	200	1600
Выбор	Пломбирование	8	300	2400
Выбор	Удаление	99	200	19800
Выбор	Пломбирование	25	300	7500
Выбор	Удаление	1	200	200
Выбор	Итого:			34200

Экспорт в Excel

Выбор	Номер зуба	Статус
Выбор	3	коронка
Выбор	5	пломба
Выбор	7	пломба
Выбор	1	пломба
Выбор	4	коронка
Выбор	11	удален
Выбор	1	винир

Рисунок 22 – Веб – форма для сотрудников

Веб – форма содержит таблицу приемов определенного сотрудника – рисунок 23.

Пациент    Приемы на:

Выбор	Пациент	Дата приема	Описание	Статус	Врач	Телефон	Email
Выбор	Николай Жирнаков	05.05.2021 0:00:00	Опухла десна	<input type="checkbox"/>	Яковлева Светлана Игоревна	8 (999) - 555 - 2392	nik@mail.ru
Выбор	Михаил Зотов	29.04.2021 0:00:00	Выпала пломба	<input type="checkbox"/>	Яковлева Светлана Игоревна	8 (999) - 554 - 2392	mix@mail.ru

Рисунок 23 – Таблица приемов для одного сотрудника

Для данной таблицы реализован поиск по пациентам и фильтрация по датам приема (сегодня, завтра, все). При выборе записи, открывается форма для внесения статуса посещения – рисунок 24.

Статус приема:

Принят

Не принят

**Ввод**

Рисунок 24 –Форма статуса приема

В таблицы, показанные на рисунке 25, загружаются данные выбранной строки приема.

Материал	Количество	Цена	Сумма
Выбор Пенцет	3	100	300
Выбор Пенцет	5	1000	5000
Выбор Пенцет	5	1000	5000
Выбор Сверло	2	2000	4000
Выбор Пенцет	9	1000	9000
Выбор Сверло	77	2000	154000
Выбор Сверло	77	2000	154000
Выбор Сверло	76	2000	152000
Выбор Итого:			483300

Лекарство	Количество	Цена	Сумма
Выбор Мышьяк	3	300	900

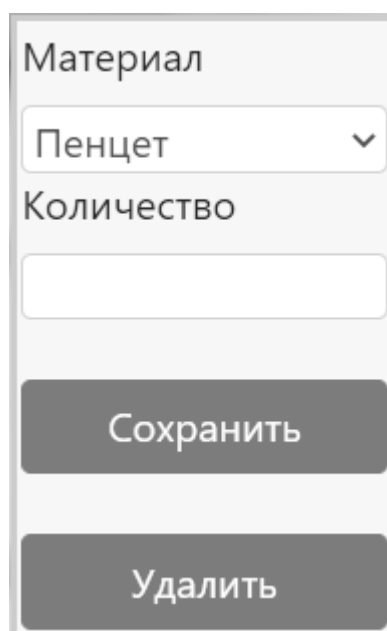
Услуга	Количество	Цена	Сумма
Выбор Удаление	3	100	300
Выбор Удаление	5	200	1000
Выбор Удаление	7	200	1400
Выбор Удаление	8	200	1600
Выбор Пломбирование	8	300	2400
Выбор Удаление	99	200	19800
Выбор Пломбирование	25	300	7500
Выбор Удаление	1	200	200
Выбор Итого:			34200

Номер зуба	Статус
Выбор 3	коронка
Выбор 5	пломба
Выбор 7	пломба
Выбор 1	пломба
Выбор 4	коронка
Выбор 11	удален
Выбор 1	винир

Рисунок 25 – Данные выбранного приема

При выборе строки одной из таблиц, открывается определенная форма выбранной таблицы, для добавления, редактирования и удаления записей – рисунок 26.



Материал

Пенцет ▾

Количество

Сохранить

Удалить

Рисунок 26 – Форма использованных материалов

В данной форме выбирается использованный материал и количество, стоимость заполняется автоматически из справочника материалов. После добавления, формируется отчет с рассчитанной суммой по каждой строке и итоговой суммой затраченных материалов – рисунок 27.

	Материал	Количество	Цена	Сумма
Выбор	Пинцет	3	100	300
Выбор	Сверло	5	100	500
Выбор	Пинцет	5	1000	5000
Выбор	Пинцет	5	1000	5000
Выбор	Сверло	2	2000	4000
Выбор	Пинцет	9	1000	9000
Выбор	Сверло	77	2000	154000
Выбор	Сверло	77	2000	154000
Выбор	Сверло	76	2000	152000

Экспорт в Excel      Довавить

Рисунок 27 – Таблица использованных материалов

Также есть возможность экспорта данного отчета в excel–рисунок 28.

Пациент	Дата приема	Описание	Врач
Николай Жирнаков	05.05.2021 0:00	Опухла десна	Яковлева Светлана Игоревна

Материал	Количество	Цена	Сумма
Пинцет	3	100	300
Пинцет	5	1000	5000
Пинцет	5	1000	5000
Сверло	2	2000	4000
Пинцет	9	1000	9000
Сверло	77	2000	154000
Сверло	77	2000	154000
Сверло	76	2000	152000
Итого:			483300

Рисунок 28 – Экспорт в excel

В отчете формируется информация о затраченных материалах определенного приема, фрагмент кода построения отчета описан в приложении Е. Аналогичный функционал имеют таблицы лекарств и услуг.

### **Выводы по второму разделу**

Выбраны инструменты решения задачи. Построена модель потоков данных (DFD), описаны бизнес – процессы стоматологической клиники (методология IDEF0) и разработана схема базы данных. Приведена инструкция создания схемы членства в SQL Server для работы с учетными данными пользователей. Описаны классы и интерфейсы программы.

## 3 ОРГАНИЗАЦИОННО-ЭКОНОМИЧЕСКАЯ ЧАСТЬ

Итоговые затраты на разработку рассчитываются методом сложения всех расходов за месяц и умножения их на количество месяцев равных сроку разработки. Расходы объединены по экономическим элементам:

- Материалы;
- Зарплата;
- Амортизация;
- Прочее.

### 3.1 Затраты на материалы

Расходы на материалы вычисляются по следующей формуле (1):

$$Z_M = \sum(q_M \times C_M) \quad (1)$$

где  $Z_M$  – затраты на материалы;

$q_M$  – количество материалов;

$C_M$  – затраты на единицу.

Затраты на материалы представлены в таблице 14.

Таблица 14– Стоимость материалов

Наименование	Единица измерения	Затраты на единицу, руб.	Количество, шт	Сумма, руб.
Канцтовары (ручка, маркер, карандаш и т. д.)	комплект	40	1	40
Бумага	пачка	130	1	130
Итого				170

### 3.2 Затраты на заработную плату

Расходы на зарплату разработчика рассчитываются по формуле (2):

$$ЗП = \Sigma( O + O \times СВ) \quad (2)$$

где ЗП – зарплата рабочих;

O – оклад в рублях;

СВ – страховые взносы (СВ = 30.2%).

Общие затраты по заработной плате разработчиков рассчитываются методом сложения затрат на заработную плату программиста разработчика и затрат на заработную плату начальника программиста разработчика, данные приведены в таблице 15.

Таблица 15 – Зарплата рабочих

Наименование	Оклад, руб	Страховые взносы, руб.	Загруженность %	Сумма, руб.
Программистразработчик	15 000	2500	80	14000
Руководительпроекта	17 000	3500	20	4100
Бизнес-аналитик	13000	2000	40	6000
Итого:				24 100

### 3.3 Затраты на амортизацию

Расчет амортизации оборудования производится по формуле (3):

$$A = \Sigma(НС/СС) \quad (3)$$

где A – амортизация;

НС – начальная стоимость;

СС – срок службы в годах.

ПК – модель ноутбука Aceraspire 1410 стоимостью 25 000 руб., срок службы 4 лет или 48 месяцев, следовательно, амортизация в месяц равна 521 руб./мес.

Программное обеспечение – MSOffice 2007. Программное обеспечение, в состав которого входит пакет MicrosoftOffice 2007 профессиональный



стоимостью 10 000 руб., срок службы данного ПО составляет 3 года или 36 месяцев, следовательно, амортизация в месяц составляет 278 руб./мес.

Расчет затрат на амортизацию представлен в таблице 16.

Таблица 16– Стоимость амортизация оборудования и ПО

Наименование	Стоимость, руб.	Срокслужбы, лет	Амортизации в месяц, руб.	Загруженность, %	Сумма, руб.
Acer aspire 1410	30 000	5	521	95	495
MSOffice 2007	10 000	3	278	90	250
MS Access 2010	бесплатно	неограничен	0	75	0
Visual Studio Community	бесплатно	неограничен	0	90	0
Итого:					745

### 3.4 Прочие затраты

В состав арендных платежей входят стоимость аренды и обслуживания помещения, бухгалтерские и прочие услуги обслуживания работы, управленческие расходы Арендные платежи рассчитываются по формуле (4):

$$A = S_m \times C \quad (4)$$

где А – Арендные платежи;

$S_m$  – площадь рабочего места (равна 5 м<sup>2</sup>);

С – затраты на аренду в месяц (за 1 м<sup>2</sup> равны 300 рублей/месяц).

$$\text{Аренда} = 5 \text{ м}^2 \times 300 \text{ руб/м}^2 = 1500 \text{ руб.}$$

Суммарные затраты на разработку Суммарные затраты за 3 месяца разработку рассчитываются по формуле (5):

$$TC = C_m \times T \quad (5)$$

где TC – суммарные затраты на разработку

$C_m$  – затраты в месяц;

T – время разработки.

Суммарные затраты на разработку системы представлены в таблице 17.

Таблица 17 – Суммарные затраты

Наименование	Затраты в месяц, руб.	Время на разработку, мес.	Сумма, руб.
Материалы	170	3	521
Зарплата	24100	3	72300
Амортизация	745	3	2235
Прочее	1500	3	4500
Итого:			79556

**Вывод по разделу три**

Рассчитаны статьи затрат на разработку, вычислена цена продукта.

## ЗАКЛЮЧЕНИЕ

В результате выполнения выпускной квалификационной работы была разработана информационная система для стоматологической клиники.

Стоматологическая клиника – это комплексное медицинское учреждение, располагающее кадрами специалистов, оснащением и оборудованием для оказания специализированной стоматологической помощи.

Основные процессы в стоматологической клинике не могут обойтись без хранения информации и ведения различной документации. Иногда возникает необходимость в формировании отчётов/списков с информацией о пациентах, их диагнозах, приёмах и использованных медикаментозных средств.

Поиск

и обработка такого объёма информации может отнимать довольно большой промежуток времени.

Зачастую, состояние информационных систем стоматологических клиник одинаковое и можно выделить основной недостаток – отсутствие возможности своевременного и быстрого анализа и мониторинга деятельности клиники ее руководством и менеджментом для осуществления дальнейшего перспективного планирования и повышения эффективности бизнеса.

Для решения поставленной задачи использованы Visual Studio 2015, SQLServer 2018, язык программирования C#, элементы htmlразметки.

В ходе выполнения дипломного проекта решены задачи:

- проектирование программного продукта;
- выбор программных средств и инструментов для разработки;
- проектирование интерфейса;
- разработка и отладка приложения;
- тестирование программного продукта;
- оценка экономической эффективности разработанного продукта.

Итогом работы над дипломным проектом является готовое приложение, которое отвечает всем предъявленным к нему требованиям. Таким образом, поставленные задачи выполнены, а цель дипломного проекта достигнута.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Автоматизация бизнес– процессов малого предприятия [Электронный ресурс]. Режим доступа: <https://moluch.ru/archive/130/36077>.
2. Автоматизация управления стоматологической организацией с использованием методов проектного и процессного управления [Электронный ресурс]. Режим доступа: <https://cyberleninka.ru/article/v/avtomatizatsiyaupravleniya-stomatologicheskoy-organizatsiey-s-ispolzovaniem-metodov-proektnogo-i-protsessnogo-upravleniya>.
3. Администратор стоматологической клиники: обязанности и функции [Электронный ресурс].Режим доступа: <https://www.gd.ru/articles/9233-qqq17-m5-17-05-2017-administrator-stomatologicheskoy-kliniki>.
4. Актуальные задачи и пути автоматизации управления бизнес – процессами – <http://www.interface.ru/home.asp?artId=20738>.
5. ГОСТ Р ИСО/МЭК 12207-2010 Информационная технология (ИТ). Системная и программная инженерия. Процессы жизненного цикла программных средств [Электронный ресурс]. Режим доступа: <https://standartgost.ru/id/1161093>.
- 6.ГОСТ34.602-1989Информационнаятехнология.Комплексстандартов наавтоматизированныесистемы.Техническоезаданиенасозданиеавтоматизированной системы [Электронный ресурс]. Режимдоступа: [https://standartgost.ru/g/%D0%93%D0%9E%D0%A1%D0%A2\\_19.201-78](https://standartgost.ru/g/%D0%93%D0%9E%D0%A1%D0%A2_19.201-78).
7. Должностная инструкция врача стоматолога N1 [Электронный ресурс]. Режим доступа: <http://www.consultpharma.ru/index.php/ru/rabota/dolzhnostnaya-instruktsiya/795-vracha-stomatologa>.
8. Должностная инструкция медицинской сестры стоматологического кабинета [Электронный ресурс]. Режимдоступа: <http://prom-nadzor.ru/content/dolzhnostnaya-instrukciya-medicinskoj-sestry-stomatologicheskogo-kabineta>.

9. Должностная инструкция медицинской сестры, должностные обязанности медицинской сестры, образец должностной инструкции медицинской сестры [Электронный ресурс]. Режим доступа: [https://www.rabota.ru/articles/hr/dolzhnostnaja\\_instruksija\\_meditinskoj\\_sestry\\_dolz\\_hnostnye\\_objazannosti\\_psihologa\\_obrazets\\_dolzhnostnoj\\_instruksii\\_psihologa-3938](https://www.rabota.ru/articles/hr/dolzhnostnaja_instruksija_meditinskoj_sestry_dolz_hnostnye_objazannosti_psihologa_obrazets_dolzhnostnoj_instruksii_psihologa-3938).
10. Должностная инструкция. Врач-стоматолог [Электронный ресурс]. Режим доступа: <https://classinform.ru/dolzhnostnye-instruktsii-profstandart/dolzhnostnaya-instrukciya-vracha-stomatolog.html>.
11. Ключевые возможности системы БИТ. Стоматология [Электронный ресурс]. Режим доступа: <https://habarovsk.1cbit.ru/1csoft/bit-stomatologiya/>.
12. Лившиц, И. «По зубам» ли дантистам системы автоматизации? [Электронный ресурс]. – Режим доступа: [http://www.cnews.ru/articles/po\\_zubam\\_li\\_dantistam\\_sistemy\\_avtomatizatsii/](http://www.cnews.ru/articles/po_zubam_li_dantistam_sistemy_avtomatizatsii/).
13. Перспективы внедрения современных технологий для совершенствования и оптимизации стоматологической службы (на примере ортодонтической стоматологии) [Электронный ресурс]. Режим доступа: <https://elibrary.ru/item.asp?id=26454186>.
14. Перспективы использования автоматизированных информационных систем в стоматологии [Электронный ресурс]. Режим доступа: <https://cyberleninka.ru/article/n/perspektivy-ispolzovaniya-avtomatizirovannyhinfor-matsionnyhsistem-v-stomatologii>.
15. Проблемы при внедрении процессного подхода к автоматизации бизнеса и механизмы их решения [Электронный ресурс]. Режим доступа: <https://cyberleninka.ru/article/n/problemy-pri-vnedrenii-protsessnogo-podhoda-kavtomatizat-sii-biznesa-i-mehanizmy-ih-resheniya>.
16. Процессное управление – это просто [Электронный ресурс]. Режим доступа: [https://www.cfin.ru/itm/bpr/process\\_management.shtml](https://www.cfin.ru/itm/bpr/process_management.shtml).

17. Руководителю предприятия. Внедрение системы автоматизации, основные проблемы и задачи. [Электронный ресурс]. Режим доступа: <https://www.cfin.ru/vernikov/kias/recommend.shtml>
18. Создание автоматизированной компьютерной системы для информационной поддержки врача-стоматолога [Электронный ресурс]. Режим доступа: <https://cyberleninka.ru/article/n/sozdanie-avtomatizirovannoy-kompyuternoy-sistemy-dlya-informatsionnoy-podderzhki-vracha-stomatologa>.
19. Что такое ClinicIQ? [Электронный ресурс]. Режим доступа: <https://www.cliniciq.ru/>.
20. DentalBase программа для автоматизации стоматологии [Электронный ресурс]. Режим доступа: <https://dentalbase.ru/>.
21. Павловская, Т.А. С#. Программирование на языке высокого уровня. / Т.А. Павловская. – Санкт Петербург: «Питер», 2007. - 544 с.

## ПРИЛОЖЕНИЕ А

### Код класса Login

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
//добавить
using System.Web.Security;

namespace _0067.Account {
    public partial class Login : System.Web.UI.Page {
        protected void Page_Load(object sender, EventArgs e) {

        }

        protected void Button1_Click(object sender, EventArgs e) {
            //проверяем наличие пользователя и пароля в базе данных
            Global.login = TextBox1.Text;
            bool userValid = Membership.ValidateUser(TextBox1.Text, TextBox2.Text);
            if(userValid) {
                //добавляем в cookies и перенаправляем страницу
                //Response.Redirect("../Administrator/Default.aspx");
                FormsAuthentication.RedirectFromLoginPage(TextBox1.Text, false);

            }
            else {
                Label1.Text = "Ошибка!";
            }
        }
    }
}
```



## ПРИЛОЖЕНИЕБ

### кодфункции populatesgridview()

```
void PopulatesGridview()
{
    DataTable dtbl = new DataTable();
    using (SqlConnection sqlcon = new SqlConnection(connectionString))
    {
        sqlcon.Open();
        SqlDataAdapter sqlDa = new SqlDataAdapter("SELECT * FROM Pa-
patients", sqlcon);
        sqlDa.Fill(dtbl);
    }
    if (dtbl.Rows.Count > 0)
    {
        GridView1.DataSource = dtbl;
        GridView1.DataBind();
    }
    else
    {
        dtbl.Rows.Add(dtbl.NewRow());
        GridView1.DataSource = dtbl;
        GridView1.DataBind();
        GridView1.Rows[0].Cells.Clear();
        GridView1.Rows[0].Cells.Add(new TableCell());
        GridView1.Rows[0].Cells[0].ColumnSpan = dtbl.Columns.Count;
        GridView1.Rows[0].Cells[0].Text = "No Data Found ..!";
        GridView1.Rows[0].Cells[0].HorizontalAlign = Horizonta-
lAlign.Center;
    }
}
```

## ПРИЛОЖЕНИЕ В

### код события RowCommand

```
protected void GridView1_RowCommand(object sender, GridViewCommandEventArgs e)
{
    try
    {
        if (e.CommandName.Equals("AddNew"))
        {
            using (SqlConnection sqlcon = new SqlConnection(connectionString))
            {
                sqlcon.Open();
                string query = "INSERT INTO Patients
(Name,Phone,Mail,Adres) VALUES (@Name,@Phone,@Mail,@Adres)";
                SqlCommand sqlCmd = new SqlCommand(query, sqlcon);
                sqlCmd.Parameters.AddWithValue("@Name",
(GridView1.FooterRow.FindControl("txtNameFooter") as TextBox).Text.Trim());
                sqlCmd.Parameters.AddWithValue("@Phone",
(GridView1.FooterRow.FindControl("txtPhoneFooter") as TextBox).Text.Trim());
                sqlCmd.Parameters.AddWithValue("@Mail",
(GridView1.FooterRow.FindControl("txtMailFooter") as TextBox).Text.Trim());
                sqlCmd.Parameters.AddWithValue("@Adres",
(GridView1.FooterRow.FindControl("txtAdresFooter") as TextBox).Text.Trim());
                sqlCmd.ExecuteNonQuery();
                PopulaterGridview();
                lblSuccess.Text = "New Record Added";
                lblError.Text = "";
            }
        }
    }
    catch (Exception ex)
    {
        lblSuccess.Text = "";
        lblError.Text = ex.Message;
    }
}
```

## ПРИЛОЖЕНИЕГ

### кодсобытия RowUpdating

```
protected void GridView1_RowUpdating(object sender, GridViewUpdateEventArgs e)
{
    try
    {
        using (SqlConnection sqlcon = new SqlConnection(connectionString))
        {
            sqlcon.Open();
            string query = "UPDATE Patients SET
Name=@Name,Phone=@Phone,Mail=@Mail,Adres=@Adres WHERE ID=@id";
            SqlCommand sqlCmd = new SqlCommand(query, sqlcon);
            sqlCmd.Parameters.AddWithValue("@Name",
(GridView1.Rows[e.RowIndex].FindControl("txtName") as TextBox).Text.Trim());
            sqlCmd.Parameters.AddWithValue("@Phone",
(GridView1.Rows[e.RowIndex].FindControl("txtPhone") as TextBox).Text.Trim());
            sqlCmd.Parameters.AddWithValue("@Mail",
(GridView1.Rows[e.RowIndex].FindControl("txtMail") as TextBox).Text.Trim());
            sqlCmd.Parameters.AddWithValue("@Adres",
(GridView1.Rows[e.RowIndex].FindControl("txtAdres") as TextBox).Text.Trim());
            sqlCmd.Parameters.AddWithValue("@id", Convert.ToInt32(GridView1.DataKeys[e.RowIndex].Value.ToString()));

            sqlCmd.ExecuteNonQuery();
            GridView1.EditIndex = -1;
            PopulaterGridview();
            lblSuccess.Text = "Selected Record Updated";
            lblError.Text = "";

        }
    }
    catch (Exception ex)
    {
        lblSuccess.Text = "";
        lblError.Text = ex.Message;
    }
}
```

## ПРИЛОЖЕНИЕ Д

### код события RowDeleting

```
protected void GridView1_RowDeleting(object sender, GridViewDeleteEventArgs e)
{
    try
    {
        using (SqlConnection sqlcon = new SqlConnection(connectionString))
        {
            sqlcon.Open();
            string query = "DELETE Patients WHERE ID=@id";
            SqlCommand sqlCmd = new SqlCommand(query, sqlcon);

            sqlCmd.Parameters.AddWithValue("@id", Convert.ToInt32(GridView1.DataKeys[e.RowIndex].Value.ToString()));

            sqlCmd.ExecuteNonQuery();

            PopulaterGridview();
            lblSuccess.Text = "Selected Record Deleted";
            lblError.Text = "";

        }
    }
    catch (Exception ex)
    {
        lblSuccess.Text = "";
        lblError.Text = ex.Message;
    }
}
```

## ПРИЛОЖЕНИЕЕ

### код события Click

```
protected void Button9_Click(object sender, EventArgs e)
{
    GridView2.Columns[0].Visible = false;
    GridView1.Columns[0].Visible = false;

    GridView1.Columns[4].Visible = false;
    GridView1.Columns[6].Visible = false;
    GridView1.Columns[7].Visible = false;

    Response.Clear();
    Response.Buffer = true;
    Response.AddHeader("content-disposition", "attach-
ment;filename=GridViewExport.xls");
    Response.Charset = "";
    Response.ContentType = "application/vnd.ms-excel";
    using (StringWriter sw = new StringWriter())
    {
        HtmlTextWriter hw = new HtmlTextWriter(sw);

        GridView2.AllowSorting = GridView2.AllowPaging = false;
        GridView1.AllowSorting = GridView1.AllowPaging = false;

        GridView2.HeaderRow.BackColor = Color.White;
        foreach (TableCell cell in GridView2.HeaderRow.Cells)
        {
            cell.BackColor = GridView2.HeaderStyle.BackColor;
        }
        foreach (GridViewRow row in GridView2.Rows)
        {
            row.BackColor = Color.White;
            foreach (TableCell cell in row.Cells)
            {
                if (row.RowIndex % 2 == 0)
                {
                    cell.BackColor =
GridView2.AlternatingRowStyle.BackColor;
                }
                else
                {
                    cell.BackColor = GridView2.RowStyle.BackColor;
                }
                cell.CssClass = "textmode";
            }
        }
        GridView1.RenderControl(hw);
        GridView2.RenderControl(hw);
    }
}
```

```
        //style to format numbers to string
        string style = @"<style> .textmode { } </style>";
        Response.Write(style);
        Response.Output.Write(sw.ToString());
        Response.Flush();
        Response.End();
    }
    GridView2.Columns[0].Visible = true;
    GridView1.Columns[0].Visible = true;
    GridView1.Columns[3].Visible = true;
    GridView1.Columns[4].Visible = true;
    GridView1.Columns[6].Visible = true;
    GridView1.Columns[7].Visible = true;
    }
```