

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

кафедра Компьютерных технологий

«Допущен к защите»
Заведующий кафедрой _____

(Ф.И.О., ученая степень, звание)

« _____ » 20__ г.

(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Разработка программного обеспечения для
медиаинформационных учреждений с использованием
технологии С#. Часть 2

Специальность БВ070400 - ВТиПО

Выполнил (а) Дю С.А. ПО-12-2
(Фамилия и инициалы) группа

Научный руководитель Туркандбаев С.С., доцент
(Фамилия и инициалы, ученая степень, звание)

Консультанты:

по экономической части:

Бекмушешев А.С., к.э.н., доцент
(Фамилия и инициалы, ученая степень, звание)
А.С. « 30 » 05 2016 г.
(подпись)

по безопасности жизнедеятельности:

Природкин И.С., д.т.н., проф.
(Фамилия и инициалы, ученая степень, звание)
И.С. « 26 » 05 2016 г.
(подпись)

по применению вычислительной техники:

Туркандбаев С.С., доцент
(Фамилия и инициалы, ученая степень, звание)
С.С. « 20 » мая 2016 г.
(подпись)

Нормоконтролер: Туркандбаев С.С., доцент
(Фамилия и инициалы, ученая степень, звание)
С.С. « 23 » мая 2016 г.
(подпись)

Рецензент: _____
(Фамилия и инициалы, ученая *степень, звание)
« _____ » 20__ г.
(подпись)

Алматы 2016 г.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Факультет Автоматических информационных технологий
Специальность Вычислительная техника и программное обеспечение
Кафедра Компьютерных технологий

ЗАДАНИЕ

на выполнение дипломного проекта

Студент Дю Елена Леонидовна
(фамилия, имя, отчество)

Тема проекта Разработка программного обеспечения для автоматизации учета энергии в котельной Ст. Тараз 2.

утверждена приказом ректора № 21 от «10» марта 2016 г.

Срок сдачи законченной работы «__» __ 20__ г.

Исходные данные к проекту требуемые параметры результатов проектирования (исследования) и исходные данные объекта

разработать БД и графическую оболочку для автоматизации работы котельных

Перечень подлежащих разработке дипломного проекта вопросов или краткое содержание дипломного проекта:

1. Определение предметной области
2. Техническое задание
3. Определение технологических средств
4. Определение специальных требований
5. Структура разработки
6. Безопасность эксплуатации
7. Экономическое обоснование

Перечень графического материала (с точным указанием обязательных чертежей)

Статистическое данные, диаграммы ИМЛ
скриншоты по коду приложения, скриншоты профилей

Рекомендуемая основная литература

1. Искусственные информационные системы. Теория и практика
конструирования Г. В. Назаренко и Р. С. Ошнова Москва 2005
2. Миттер Рад, Карлос Корона. Системы баз данных: проектирование,
реализация и управление, 5-е издание Санкт-Петербург, 2004
3. Акимович А. К., Яшев. Информационные системы
в медицине

Консультанты по проекту с указанием относящихся к ним разделов

Раздел	Консультант	Сроки	Подпись
БЖСА	Природов И. Г.	18.03 - 26.05.16	<i>[Подпись]</i>
Эконом. часть	Бекмезе А. Ч.	19.03 - 30.05.16	<i>[Подпись]</i>
Нормоконтроль	Журанов В. С.	19.05 - 23.05.16	<i>[Подпись]</i>
Основная часть	Журанов В. С.	21.03 - 05.05.16	<i>[Подпись]</i>

Аннотация

В настоящей дипломной работе представлено программное обеспечение, предназначенное для оптимизации работы медицинских учреждений. Данный программный продукт был разработан с использованием языка программирования C#.

Данный дипломный проект также содержит раздел безопасности жизнедеятельности, где проведен анализ условий труда программиста при разработке и реализации программного обеспечения.

Имеется также экономическое обоснование проекта, которое содержит подробный подсчет расходов в ходе разработки, а также сопровождении рассматриваемой программы.

Аңдатпа

Бұл бағдарламалық қамтамасыз ету медициналық мекемелердің жұмыстың жақсарту үшін арналған, осы дипломдық жұмыста ұсынылған. Осы бағдарламалық өнімі C# деген программалау тілімен әзірленді.

Дипломдық жобасының құрамында тіршілік қауіпсіздігі бөлімі бар. Бұл бөлімінде бағдарламаның еңбек шартының талдауы жасалынған.

Сонымен қатар жобасының экономикалық дәлелделі бар, оның құрамына шығыстарының есептеуі кіреді.

Abstract

In this thesis work presents a software designed to optimize the work of medical institutions.

This software product was developed using the C# programming language.

This thesis project also contains a section where the life safety analysis of working conditions of the programmer in software development.

There is also an economic justification of the project, which contains a detailed calculation of costs during development.

Содержание

Введение.....	4
1 Определение предметной области.....	5
1.1 Назначение и принцип работы ПО.....	6
1.2 Цели создания медицинского ПО.....	7
1.3 Обзор существующих аналогов.....	7
2 Техническое задание.....	9
2.1 Основные задачи.....	9
2.2 Требуемые функциональные возможности.....	9
2.3 Общие технологические требования к ПО.....	10
2.4 Исходные данные.....	10
3 Средства разработки ПО.....	11
3.1 Обоснование выбора среды разработки.....	11
3.1.1 Сравнение MS SQL Server с Oracle.....	11
3.2 Дatalogическая модель.....	19
3.3 Требования к техническому обеспечению.....	20
4 Разработка БД и описание интерфейса приложения.....	21
4.1 Анализ предметной области.....	21
4.2 UML диаграммы.....	24
4.2.1 Диаграмма прецедентов.....	25
4.2.2 Диаграмма компонентов.....	26
4.2.3 Диаграмма классов.....	26
4.2.4 Алгоритм работы программы.....	27
4.3 Проектирование БД.....	29
4.3.1 Концептуальное проектирование.....	29
4.3.2 Разработка и построение подробной ER – диаграммы на основании бизнес-правил.....	29
4.3.3 Определение атрибутов каждой сущности.....	33
4.3.4 Расчет места для хранения БД.....	35
4.4 Логическое проектирование.....	36
4.4.1 Меры по обеспечению безопасности.....	36
4.4.2 Создание пользователей, логинов и задание им паролей.....	36
4.4.3 Создание таблиц.....	37
4.5 Описание интерфейса приложения. Работа с приложением.....	39
5 Экономическое обоснование.....	45
5.1 Описание работы и обоснование необходимости.....	45
5.2 Трудовые ресурсы используемые в работе.....	45
5.3 Оборудование используемое в работе.....	45
5.4 Программное обеспечение используемое в работе.....	46
5.5 Сроки реализации проекта.....	47
5.6 Расчет стоимости работы по разработке.....	47
5.6.1 Расчет затрат на оплату труда.....	48
5.6.2 Расчет затрат по социальному налогу.....	50
5.6.3 Расчет амортизационных отчислений.....	51
5.6.4 Расчет затрат на электроэнергию.....	52
5.6.5 Расчет накладных и прочих расходов.....	53
5.6.6 Расчет стоимости по всем статьям затрат и определение структуры затрат.....	54
5.7 Цена интеллектуального труда.....	55
5.8 Вывод.....	56
6 Безопасность жизнедеятельности.....	57
6.1 Анализ условий труда.....	58
6.2 Расчетная часть.....	62

6.2.1 Расчет уровня шума.....	62
6.3 Расчет молниезащиты офисного здания одиночным стержневым молниеотводом.	64
6.4 Вывод.....	66
Заключение.....	67
Список литературы.....	68
Приложение А.....	69

Введение

В современном мире все общество компьютеризировано, это гениальное устройство, предпосылки для изобретения которого были еще в Древней Греции, теперь используется почти во всех сферах деятельности и, практически, повсеместно. Каждая развивающаяся или уже развитая страна идет в ногу со временем, и с каждым днем на столах крупных, или не очень, организаций, государственных учреждений и т.д. появляется все больше персональных компьютеров различных видов, комплектаций и марок.

Персональные компьютеры охватили и область медицины. С помощью таких машин производятся лабораторные исследования, обследования пациентов и многое другое. Мало просто иметь такую электронно-вычислительную машину, самым важным моментом является подбор качественного программного обеспечения, способного улучшить производительность работы того или иного медицинского учреждения.

Для начала разберемся что же такое программное обеспечение. Программное обеспечение, сокращенно ПО, происходящее от английского слова software или, по-другому, называют – «софт», является одной из самых важных и быстроразвивающихся составляющих технологий информации современности, систем-автоматов, мировых информационных ресурсов. Программное обеспечение делится на операционную систему, которая обеспечивает функционирование и взаимосвязь всех компонентов компьютера и предоставляет пользователю доступ к его аппаратным возможностям, и прикладное программное обеспечение, которое, в свою очередь, делится на средства разработки, являющиеся инструментами программиста, и приложения. В настоящем дипломном проекте рассматривается приложение для типовых поликлиник.

Приложение – программные продукты, либо пакеты прикладных программ, ориентированные на непрограммирующего пользователя и реализующие определенные функции, такие как, работа с документами, мультимедийными материалами, коммуникация и т.д.

На сегодняшний день государственные медицинские учреждения хранят все данные о пациентах, включая истории болезней, приемов и так далее, в бумажном виде, не удивительно, что многие люди долгое время, а иногда и вовсе не могут найти свои амбулаторные карточки, при этом пациентам приходится часами ждать своей очереди в несколько кабинетов. Порой одно лишь обследование человек проходит несколько дней. Вышесказанное, отчасти, является причиной того, что люди, просто-напросто, не хотят посещать медицинские заведения, и все чаще прибегают к малоэффективной народной медицине либо откладывают прием на неопределенный срок, вскоре совсем оставляя в стороне вопрос о своем здоровье.

Программный продукт, рассматриваемый в настоящем дипломном проекте, преследует цель - оптимизировать работу поликлиник посредством ускорения процесса обследования и сбора необходимых анализов, наладить

взаимодействие между отделениями поликлиники и лаборатории, с помощью создания корпоративной сети. Необходимость своевременного обследования и лечения больных, сокращения количество очередей, затрат денег и времени на прием, необходимость использовать большое, и по-прежнему неуклонно растущее, количество информации при решении диагностических, терапевтических, статистических, административных и других задач, определяет создание сегодня программного продукта в медицинских учреждениях.

1 Определение предметной области

В более широком смысле само понятие «система», в какой-то мере, можно определить, как совокупность взаимосвязанных элементов. Это понятие, в последующем, будет использоваться в более узком смысле. С точки зрения прикладной науки, системой является совокупность средств, организованных по определенному принципу для выполнения поставленной задачи.

Конкретная система может характеризоваться с точки зрения:

- решаемой задачи;
- средств разработки, необходимых для решения поставленной задачи;
- процессов (алгоритмов) преобразования поступающих исходных данных в желаемый результат.

Под информационной системой понимается комплекс методологических, программных, технических, информационных, а также организационных средств, поддерживающих процессы функционирования информатизируемой организации.

Если система выполняет определенные функции самостоятельно (без участия человека) или нет, т.е. если человек является составляющим этой системы, то информационная система может быть автоматической или автоматизированной.

Программный продукт доведенный до автоматизации позволяет выполнять как ручные, так и компьютерные процессы — пользователь (оператор) и компьютерные средства совместной работы с целью обработки и дальнейшего использования входной информации.

Так как лечебно-диагностический процесс невозможен без воздействия человека (врача), в дальнейшем мы будем рассматривать лишь автоматизированные компьютерные системы, либо просто автоматизированные системы.

Назовем компьютеризацией лечебного и обследовательского процесса - внедрение в медицинскую практику и развитие компьютерного аппаратно-программного обеспечения.

Медицинская автоматизированная информационная система или сокращенно (МИС) – это есть совокупность программно-технических средств, а также базы данных и базы знаний, ориентированных на автоматизацию различных процессов, протекающих в лечебно-профилактическом учреждении.

Автоматизированное рабочее место (АРМ) — рабочее место врача, оснащенное вычислительной техникой и программным обеспечением для сбора, хранения, обработки медицинской информации, примененное им в роли инструмента прогнозирования в процессе принятия лечебных и диагностических решений.

1.1 Назначение и принцип работы ПО

Идея заключается в том, чтобы открыть кабинет для выпускников медицинских ВУЗов с дипломом врача, кабинет будет оборудован 6-ю – 7-ю столами, каждое рабочее место будет оснащено компьютером, с данным программным обеспечением, а также стандартным набором инструментов врача. При первичном осмотре будут вбиваться данные, по которым система выдаст список анализов которые нужно сдать. Результаты будут фиксироваться в единой базе, откуда лечащий врач сможет извлечь информацию, затем поставить окончательный диагноз и назначить соответствующее лечение.

Данное ПО позволит сократить затраты пациентов, т.к. не будет необходимости оплачивать прием дважды (первичный и вторичный). Еще одним преимуществом является то, что все медицинские карты будут храниться в единой базе, доступ к которым будет открыт для тех медицинских работников, которые имеют соответствующие права на просмотр и редактирование информации. Упрощение системы первичного осмотра пациентов позволит снизить затраты времени на каждого больного.

Преимущества:

- программный продукт является бесплатным;
- экономия времени и денег пациентов;
- повышение качества работы медицинских учреждений;
- трудоустройство молодых специалистов.

1.2 Цели создания медицинского ПО

Главная цель - помочь людям получить желаемую качественную и, в то же время, доступную медицинскую помощь. Для того, чтобы достичь этой цели, выстраивается соответствующая подцель – оптимизирование работы лечебно-профилактического учреждения.

1.3 Обзор существующих аналогов

1) МИС MGERM

Интеграция МИС MGERM с кассовым аппаратом

Печать, Эл. почта 



В МИС MGERM появился модуль фискального регистратора.

Зачем он нужен?

- Кассиру не нужно вбивать сумму вручную: после закрытия счета в МИС кассовый аппарат автоматически распечатывает чек;
- Руководитель может быть спокоен: исключается возможность совершения ошибок, а также махинаций со стороны недобросовестных кассиров;
- Клиент получает информативный кассовый чек с перечислением всех оказанных услуг.

Как это выглядит?

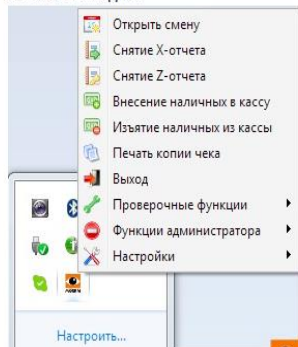


Рисунок 1 – МИС MGERM

В данной информационной системе акцент сделан на удобство оплаты медицинских услуг, идея неплохая, выбираем в меню необходимую функцию и далее печатается чек, но, по-моему это никак не способствует снижению затрат пациентов и качеству оказания помощи.

2) МИС Медиалог

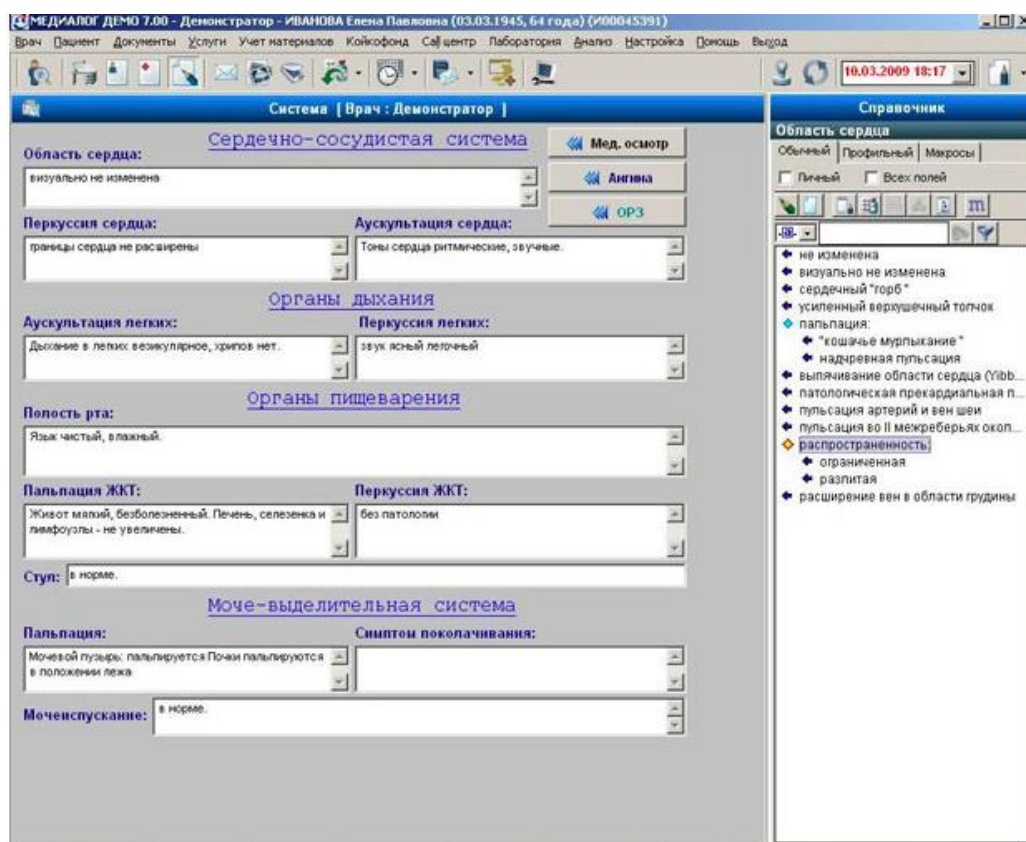


Рисунок 2 – МИС Медиалог

На Рисунке 2 представлена информационная система под названием «Медиалог».

Примерные возможности:

- Позволяет пользователям быстро и легко вводить информацию о пациенте.
- Обеспечивает безопасный доступ к правам доступа пользователей на основе ЭМК к медицинской информации, утвержденной медицинским учреждением.
- Позволяет просматривать ЕНР пациента и быстро найти информацию в больших объемах медицинской документации.
- В системе могут быть сформированы на основе ЭМК разнообразные справки, ссылки, истории болезни, существует возможность распечатать их и хранить копии этих документов.
- Дает возможность графически просматривать медицинскую информацию о пациенте: диагноз, список назначений и тому подобное.
- Предоставляет возможность подключать различные документы к ЭМК, такие как голосовое сообщение.
- Позволяет в электронном виде передать ЭМК пациенту например на флэш-накопителях.

Очень удобная система, действительно позволяющая оптимизировать работу медицинского учреждения. Однако данное программное обеспечение используется на территории Российской Федерации, следовательно преимуществом нашей программы является то, что она будет использоваться на территории нашей страны, где аналогов данного продукта практически нет.

2 Техническое задание

2.1 Основные задачи

Исходя из главной цели разработки корпоративной МИС, выстраивается ряд соответствующих взаимосвязанных задач, среди которых следующие:

- создать единую информационную базу, и, как следствие, мы добьемся ускоренного доступа к информации, а также повысится качество мед.документации и так далее;
- производить мониторинг и управлять качеством медицинской помощи, результатом чего может стать снижение вероятности врачебной ошибки, а также возможно устранение избыточности назначений и многое другое;
- деятельность того или иного медицинского учреждения станет более прозрачной;
- сокращение сроков обследования и лечения.

2.2 Требуемые функциональные возможности

Задачи, приведенные выше, обуславливают диапазон функциональных возможностей медицинской информационной системы. Основными возможностями являются:

- сбор, запись, структурирование и документирование данных;
- обеспечение обмена информацией и создания информационного пространства;
- создание единого информационного пространства, а также обеспечение обмена информацией
- надежное хранение и быстрая навигация по информации;
- исследование данных по статистике;
- слежение за эффективностью и качеством предоставления медицинских услуг;
- поддержка при принятии решения;
- анализ и учет деятельности медицинского центра, контроль запасов медицинского центра;
- ведение экономики хода лечения;
- учеба работников.

2.3 Общие технологические требования к ПО

Таблица 1. Общие требования к ПО

Требование	Обеспечиваемые преимущества
Интеграция информационных потоков	Единое информационное пространство. Координация процессов управления и хранения информации.
Синтез централизованных и распределенных технологии	Гибкость процессов управления данными: централизованный доступ к информации и распределенная инфраструктура программного продукта (в соответствии с физической структурой медицинского учреждения).
Архитектура компонентов	Позволяет делить систему на более простые ситемы.
Открытость и поддержание стандартов	Понижение объема работы над разработкой . Предоставление удобства и адаптивности системы. Приминение навыков для снижения возможности появления недочетов.
Масштабируемость и переносимость	Возможность соответствия программного продукта требованиям заказчика.
Надежность и отказоустойчивость системы	Малая вероятность отказа системы во время работы, и безопасное восстановление.
Предоставление безопасности и конфиденциальности информации	Ответственность за предоставлении конфиденциальности в границах всей медицинской системы.

2.4 Исходные данные

В роли исходных данных выступают различные виды анализов, медикаментов, симптомы и болезни, также база данных должна включать в себя полные данные о пациентах. Также данные о приемах и врачах будут являться исходными.

3 Средства разработки ПО

3.1 Обоснование выбора среды разработки

Основные запросы к создаваемому программному продукту являются постоянность, масштабируемость, многопользовательская поддержка, совместимость, полнота.

Выбором развития информационной системы окружающей среды является определение требований программного продукта и наиболее подходящей технологии для их решения.

Microsoft SQL Server

Наиболее важные характеристики базы данных - это:

- простота администрирования;
- возможность подключения к Интернету;
- скорость и функциональность механизма сервера баз данных
- наличие средств удаленного доступа

В наборе означает введение базы данных включает в себя комплект необходимых мастеров и автоматического настраивания параметров конфигурации. Кроме того, эта база данных имеет замечательное средство репликации для синхронизации данных ПК в базу данных информации и наоборот. Прилагаемое сервер OLAP предоставляет возможность хранить и анализировать все имеющиеся данные пользователя. В принципе, эта база данных представляет собой модернизированную многофункциональную базу данных, которая точно подойдет как для малой, так и для средней организации.

Следует отметить, что в SQL Server считается уступает другим СУБД по двум важным факторам: программируемость и инструментов для работы. При разработке базы данных клиентов на основе Java-приложений, HTML часто является проблемой отсутствия программного обеспечения SQL Server и использовать эту базу данных будет сложнее, чем Oracle или Sybase. Глобальная тенденция в XXI веке, стало почти повсеместным переходом к платформе Linux, и SQL Server работает только на Windows. Поэтому рекомендуется использовать SQL Server, на наш взгляд, только тогда, когда доступ к содержимому базы данных используется исключительно ODBC стандарт, в противном случае лучше использовать другие базы данных.

3.1.1 Сравнение MS SQL Server с Oracle

Бизнес-аналитика

SQL Server 2008 имеет продвинутые способности бизнес аналитики, более продвинутые в сравнении с немалыми способностями SQL Server. Несмотря на то, что решение Oracle не так популярны, так как SQL Server, используйте аналогичную функциональность будет стоить 800% дороже в Oracle. Gartner отказывается от Microsoft ставит в первую очередь на способность успешно реализовать задумки в рыночную реальность. Исходя

из данных отчета Gartner, в SQL Server 2008 возможности бизнес-аналитики были существенно увеличены и стали лучше, что позволит довести мощность бизнес-аналитики на рабочем столе каждого сотрудника организации.

Встроенные средства бизнес-аналитики

SQL Server 2008 включает в себя полностью интегрированное решение для бизнес-аналитики, для которых нет дополнительных расходов. Продукт обеспечивает поддержку хранилища данных корпоративного класса, оперативной аналитической обработки данных (OLAP), создание репортов, совокупности показателей, интеллектуального анализа данных, ETL-операции и ключевых показателей эффективности (KPI).

В сравнении с Oracle эти солушены полностью интегрированы, так что их развитие, контроль, планирование и развертывание осуществляются с применением типичных инструментов SQL Server.

Интеграция с системой Microsoft Office

Система Microsoft Office де-факто стала стандартными средствами повышения эффективности. Интеграция данных SQL Server с системой Microsoft Office может провести мощный анализ данных для сотрудников, сократить время разработки, уменьшить обучение и расходы на поддержание и улучшить эффективность работы. Реализация такой интеграции, что стоит бесплатно с SQL Server 2008 в случае Oracle потребовалось бы около \$ 30 000 за процессор расходов.

Надстройки интеллектуального анализа данных для Microsoft

Надстройки для данных Office 2007, которую можно загрузить бесплатно, улучшают интеграцию между системами Microsoft Office и SQL Server 2008. Они включают в себя анализ данных, прогнозы, калькулятор, инструменты для анализа корзины покупок, перекрестной проверки и отчетности.

Самый быстроразвивающийся поставщик средств бизнес-аналитики

В отчете IDC на долю на мировом рынке поставщиков аналитических инструментов, опубликованных в июне 2007 года Microsoft назвал наиболее быстро растущим из 10 лучших поставщиков инструментов бизнес-аналитики. Темпы роста Microsoft дважды опережают темп роста Oracle и Hupregion, а доля рынка в два раза больше, чем у Oracle, и в половину - чем Гипериона.

Совокупная стоимость владения

SQL Server не только имеет меньшую стоимость лицензии, чем Oracle 11g, но и из-за повышения уровня устойчивости, надежности и производительности, обеспечивает быструю окупаемость инвестиций. Кроме того, SQL Server 2008 имеет такие аналогичные возможности, которые в Oracle будет стоить значительно дороже, с точки зрения процессора, и что было бы необходимо приобрести дополнительные, в дополнение к базовой лицензии Oracle.

Стоимость лицензии является не единственным моментом, который следует учитывать при сравнении систем. Совокупная стоимость владения является цена, которую вы платите в конце концов, а порой системы с минимальной ценой лицензии имеют наибольшую полную цену владения. Из-за многих скрытых затрат лицензионного сбора Oracle больше, чем SQL Server, в данном случае с SQL Server включает в себя много необходимых инструментов, которые не требуют дополнительной оплаты.

Расширенные возможности администрирования

Пакет SQL Server 2008 включает в себя набор современных инструментов управления, таких как Management Studio, Performance Studio, средства управления на основе политик и PowerShell. Доступ ко всем этим инструментам, за исключением оболочки SQL Server PowerShell с помощью знакомых стандартных интерфейсов.

Надстройки для Oracle Enterprise Manager призваны обеспечить аналогичные возможности, но они требуют дополнительных лицензий.

Сокращение трудозатрат

Независимый аналитик фирмы Alinean пришел к выводу, что в среднем один администратор базы данных может управлять базами данных SQL Server 30 в то время как в случае Oracle Database один учетной записи администратора для 10 баз данных. В результате, годовая стоимость администрирования SQL Server составляет \$ 2847 по сравнению с \$ 10206 в Oracle.

Сокращение времени создания решения и более высокая производительность

Как уже упоминалось ранее, тесная интеграция с платформой приложений Microsoft и передовых средств создания программ (с использованием LINQ и ADO.NET Entity Framework) повышают производительность труда разработчиков. Кроме всего, платформа Microsoft.NET является приоритетной средой создания программ в отличие от J2EE.

Данные достоинства приводят к улучшению эффективности работы программ, быстрой постройке решений, возможно повысить надежность, снизить расходы на поддержку и, тем самым, снизить совокупную стоимость владения.

Отличная СУБД для продуктов SAP

SQL Server представлена как лучшая платформа базы данных для SAP продуктов. Пояснения содержатся в информационном документе.

SQL Server остается платформой с минимальной совокупной ценой владения для SAP продуктов. В анализе, представленном независимой компанией Wipro Technologies и охватывающих 68 SAP / ERP клиентов, пишется, что «Миграция Microsoft SQL Server предоставляет большие достоинства для SAP / ERP клиентов». В исследовании делается вывод, что "перевод SAP / ERP среды на платформу SQL Server приведет к понижению внеплановых простоев на 20% ... сократить ИТ-затраты на рабочую силу почти на 25% ... сокращению затрат на постоянную поддержку программного обеспечения на 85%". Для получения дополнительной информации см. В исследовании.

Вся функциональность «в одной коробке»

SQL Server включает в себя все функции, необходимые для принятия решения для работы с базами данных и данных на уровне организации; Oracle подвергает предприятия дополнительным затратам за аналогичные функции. На приведенном ниже рисунке показано сравнение стоимости SQL Server 2008 и Oracle 11g для стандартного четырехъядерного сервера с одним процессором. Если цена SQL-сервера не зависит от количества ядер в процессоре.

Производительность и масштабируемость

Все чаще SQL Server принимается как один из самых быстрых и масштабируемых средств управления базами данных. SQL Server 2008 показал большую производительность при выполнении тестовой отрасли. В этой версии воплощено множество улучшений в масштабируемости.

Тесты

SQL Server 2008 уже побила много рекордов в отраслевых и партнерских тестированиях. Все чаще эта база данных выбрана для программ с высокой производительностью и масштабируемых систем.

Организация Transaction Processing Performance Council

Деятельность некоммерческой организации **Transaction Processing Performance Council** (TPC) заключается в определении обработки

транзакций тестирования и эффективности работы баз данных. TPC публикует объективные цифры эффективности работы, полученные в этих тестах. TPC тесты накладывают чрезвычайно жесткие запросы к надежности и отказоустойчивости и проходят независимый аудит.

- Некоммерческая организация обработки транзакций Совет Performance (TPC) была создана для определения эталонные обработки транзакции и баз данных.

- Новый масштабируемый тест производительности TPC-E предназначен для представления современной системы оперативной обработки транзакции (OLTP). В этом тесте, в отличие от его предшественника TPC-C, использует сложную, но реалистичную схему базы данных, и это требует базовые возможности, такие как целостность данных и RAID-хранилище.

- Поддержка принятия решения TPC-H тест состоит из серии специальных требований и одновременных изменений данных, характерных для отрасли в целом.

- По состоянию на 14 апреля 2008 SQL Server 2008 продемонстрировал в тесте TPC-E рекордного количества - 1126 транзакций в секунду. SQL Server превзошел Oracle 11g в тесте TPC-H с точки зрения цены и производительности в объеме 100 ГБ, 300 ГБ, 1 ТБ и 3 ТБ.

- Компания Oracle неоднократно заявляла свои лучшие результаты в тесте соотношения цены и производительности TPC-C, TPC-E в тесте лучше отражает потребности клиентов. До этого все 10 лучших результатов в тесте TPC-C с точки зрения цены и производительности принадлежали SQL Server. Oracle результат достигается за счет использования ниши лицензирования и варианты поддержки, которые мало применимы к реальной жизни. В случае Oracle была использована лицензия на Oracle 11g, действительна только в течение трех лет, а в случае SQL Server - лицензия с неограниченным сроком действия. Изначально, поддержка Oracle бесплатна, но схема лицензирования, используемый в тесте, включает в себя один платеж каждого вызова в поддержку. Кроме того, Oracle использовала продукт Standard Edition One, для которых практически нет спроса со стороны корпоративных клиентов. В случае SQL Server, его лучшие результаты с точки зрения цены и производительности достигаются на Enterprise Edition 64-разрядной версии, типичной для корпоративного сегмента.

Тесты партнеров

SQL базы данных сервера показал отличные результаты в тестах различных партнеров.

- SQL Server 2008 продемонстрировала рекордные показатели в 3-уровневой тестов SAP продаж и дистрибуции (SD) Standard Application на сервере 4-процессорной из категории обычного широкопоставленных блейд-серверов и с участием 34000 пользователей SAP SD Standard Application.

- SQL Server 2008 достиг исключительную ценность и производительность.

- Компания Unisys добилась мирового рекорда в области извлечения, преобразования и загрузки (ETL), выполняя загрузки 1 терабайта данных меньшеб чем за полчаса времени. Данный рекорд был установлен интеграцией SQL Server 2008 Integration Services.

- Компания Camstar, ведущий поставщик автоматизированных систем управления технологическими процессами производства (MES) для глобальных предприятий, сегодня объявила о мировой рекорд - 205 MES транзакций в секунду. В котором скорость обработки увеличивается на 14%, а данные заполняемого пространства снизились на 60% за счет уменьшения объема базы данных. Эти результаты были достигнуты с применением Camstar, SQL Server 2008 и Windows Server® 2008, когда по сравнению с SQL Server 2005.

- Microsoft Dynamics AX сообщает запись (70%) улучшение производительности, масштабируемости и отклика системы время ссылки на контрольные показатели .В показали рекордные результаты - производительность, масштабируемость и время отклика улучшилось на 70%. Увеличение эффективности работы сопровождался сокращением увеличения объема базы данных за счет снижения объема базы данных в SQL Server 2008.

- Microsoft Dynamics CRM сообщает о рекордных цифрах - 24 000 одновременно работающих пользователей, а время отклика была доля секунды. Запись производительности в тестах с 24000 одновременно работающих пользователей было достигнуто с помощью Microsoft Dynamics CRM 4.0, SQL Server 2008 и Windows Server 2008 для корпоративных рабочих нагрузок.

Поддержка независимых поставщиков программного обеспечения

Учитывая тот факт, что в настоящее время операционная система Windows Server 2008, более чем две трети продаж новых серверов, а также базы данных SQL Server стала одной из самых топовых систем управления базами данных, независимые поставщики программного обеспечения все чаще выбирают платформу SQL Server под управлением Windows.

Примеры внедрения независимыми поставщиками ПО

Следующие примеры показывают поддержку SQL Server независимыми поставщиками программного обеспечения.

Siemens тестировала свое программное обеспечение для управления жизненным циклом изделия (PLM) для SQL Server 2008 с 5000 пользователей. Сравнение SQL Server 2005 показал следующие результаты:

- Уменьшение размера файлов базы данных на 50% при использовании сжатия;
- Сокращение времени отклика на 20%;
- Улучшенная масштабируемость;
- Снижение загрузки процессора на 10%;
- сокращение использования ОЗУ на 5%.

RedPrairie Компания объявляет о смене настроении клиентов: 95% клиентов, ранее применяемых для решения о UNIX на основе, в настоящее время 70% клиентов просят решения, основанные на Windows. По оценкам компании, использование платформы приложений Microsoft обеспечивает более чем вдвое снизить стоимость развертывания по сравнению со стоимостью аппаратных средств и программного обеспечения на UNIX основе solutions.The Следующие примеры показывают поддержку SQL Server независимыми поставщиками программного обеспечения.

Siemens тестировала свое программное обеспечение для управления жизненным циклом изделия (PLM) для SQL Server 2008 с 5000 пользователей. Сравнение SQL Server 2005 показал следующие результаты:

- Уменьшение размера файлов базы данных на 50% при использовании сжатия;
- сокращение времени отклика на 20%;
- Улучшенная масштабируемость;
- Снижение загрузки процессора на 10%;
- сокращение использования ОЗУ на 5%.

RedPrairie Компания объявляет о смене настроек клиентов: 95% клиентов, ранее применяемых для решений о UNIX на основе, в настоящее время 70% клиентов просят решения, основанные на Windows. По оценкам компании, использование платформы приложений Microsoft обеспечивает более чем вдвое снизить стоимость развертывания по сравнению со

стоимостью аппаратных средств и программного обеспечения на UNIX-решений на базе.

Безопасность

Безопасность является важным фактором в деле защиты интеллектуальной собственности и сохранения доверия клиентов и партнеров. SQL базы данных сервера подтвердила свою позицию лидера рынка в области безопасности баз данных. По сравнению с базой данных Oracle 11g, которая имеет меньше уязвимостей, обладает надежными функциями безопасности, которые не требуют дополнительную плату, и гораздо лучше, система управления исправлениями.

Средства безопасности

Контроль на основе политик предоставляет активное применение политик в отношении объектов базы данных. Политики содержат набор условий, которые могут быть использованы для применения бизнес-правил и правил техники безопасности.

Защищенные информационные системы

Концепция защищенных компьютерных систем от Microsoft (Trustworthy Computing) гарантирует надежность и безопасность программного обеспечения и услуг корпорации, уважение конфиденциальности пользователей, и при поддержке надежных и ответственных компаний.

Согласно Enterprise Strategy Group аналитик фирмы (ESG), концепция «безопасного» жизненного цикла разработки Microsoft (Security Development Lifecycle, SDL) является лидером в области безопасности и должны быть приняты как можно скорее принять независимых поставщиков программного обеспечения.

Критические уязвимости системы безопасности

Национальная база данных США об уязвимостях (NVD) зарегистрировано более 250 критических уязвимостей в системе безопасности, найденные в базах данных Oracle в течение последних 4-х лет. За тот же период, что не сообщалось о каких-либо слабых местах SQL Server. База данных NVD - публичное хранилище информации об уязвимостях на основе стандартов США, которые обслуживаются Национальным институтом стандартов и технологии.

Инфраструктура управления обновлениями

Обновление инфраструктуры Microsoft - простое современное решение для управления обновлениями.

Решения Oracle для установки патчей настолько сложна, что, по мнению Computerworld, «две трети администраторов баз данных Oracle не применяются патчи безопасности." В той же статье говорится о "мучительной боли", связанной с установкой патчей Oracle.

Согласно InfoWorld, Oracle на пять лет отстает Microsoft в управлении обновлениями.

Ограничение доступа к данным для пользователей с высоким уровнем привилегий

Если некоторые пользователи обладают высокими привилегиями, это не означает, что они должны иметь доступ ко всем данным. Например, вряд ли предоставит старший базы данных администраторам доступ к данным о финансовых и кадровых данных.

Используя новые возможности аудита, предоставления индивидуальных разрешений пользователям, модуль подписи, TDE, HSM и управления на основе политик в SQL Server позволяет предотвратить несанкционированный доступ к конфиденциальным данным со стороны пользователей с повышенными привилегиями. Все эти особенности SQL Server 2008 доступны без дополнительной оплаты.

Oracle Database Vault включает в себя компонент для доступа к привилегированному управлению пользователями. Это стоит \$ 20 000 на процессор и документации Oracle четко говорится, что «компонент DB Vault не предотвращает прямой доступ к данным со стороны пользователи с повышенными привилегиями."

Резюмирую.

1) ORACLE и SQL SERVER позволяют оперировать огромными данными (в терабайт), но ORACLE большие объемы данных гораздо более продуктивным, чем SQL SERVER, с другой стороны, есть еще такие потребности при работе с большим количеством данных, которые не были бы справились SQL Server.

2) Существуют различия в платформе выбора. Обе системы могут работать на Windows NT, но в то же время, как и для SQL SERVER является родной платформой, существует целый ряд плюсов для него. ORACLE также может работать на платформе UNIX и использовать свои возможности.

3) SQL SERVER а) система безопасности может быть настроена в соответствии с системой безопасности Windows NT. б) Тесная интеграция как современные инструменты разработки приложений, а также для настольных приложения, таких как Microsoft Word, Access и Excel. в) доступ

к базам данных SQL Server можно с помощью Microsoft Jet Engine, а объекты доступа к данным (DAO), удаленных объектов данных (РДУ), объекты данных ActiveX (ADO), OLE DB, ODBC, встроенный SQL Server Library (DB-Library) и другие технологии, разработанные третьими лицами (что может быть большой минус).

4) В этих системах различные типы данных реализованы, но это лишь вопрос удобства и привычки, но на ORACLE, разработанный ANSI стандарт языка PL / SQL упрощает выполнение многих задач.

5) Обе системы разработаны способы перемещения по записям и работать с ними. В SQL SERVER сделан более гибким, то есть движение в любых записях и снимать работу. В ORACLE возможно только скидка с текущего счета и назначение следующего.

Таким образом, принимая во внимание фактор оборудования единиц компьютерной техники, было принято решение:

1. О разработке базы данных на основе MS SQL Server, который из-за своей современной, свободной и конкурентоспособность по сравнению с аналогичными продуктами. В MS SQL Server реализованы множество функций, как правило, можно найти только в коммерческих базах данных, таких как DB2 и Oracle. Простота расширения - в MS SQL Server поддерживаются пользовательские операторы, функции, методы доступа и типы данных. Языки, называемые процедурными - обеспечивает поддержку внутренних процедурных языков, включая PL / PgSQL, который является аналогом специалиста языка PL / SQL, Oracle процедурного языка. Одно из преимуществ MS SQL Server является возможность использования Perl, Python и TCL в качестве внутренних процедурных языков.

2. Развитие графической среды для MS-студия 2013. визуальной разработки системы Microsoft Visual Studio® можно использовать для разработки приложений SQL Server, включая клиента и промежуточного применения и приложения уровня данных. Visual Studio интегрирована с системами управления жизненным циклом, тест-систем, серверных продуктов Microsoft и системы Microsoft Office.

3.2 Даталогическая модель.

Даталогическое дизайн - создание схемы базы данных на основе конкретной модели данных, такие как реляционной модели данных. Для реляционной модели данных даталогической модели - набор схем отношении, как правило, с первичным ключом, а также отношения между отношениями, которые являются внешними ключами.

Для даталогических моделей, определенных методологии построения

графиков, IDEF.- методология ICAM семьи (Integrated Computer-Aided Manufacturing) систем для решения сложных задач моделирования, позволяет отображать и анализировать активность широкого спектра моделей сложных систем в различных разделах. Эта широта и глубина процессов обследования в системе определяется разработчиком, что позволяет не перегружать создавать модели избыточных данных. IDEF1X (IDEF1 Extended) - методология построения реляционных структур (баз данных), относится к типу отношений методологии Сущность и обычно применяется для проектирования реляционных баз данных, относящихся к этой системе.

3.3 Требования к техническому обеспечению

Определение вычислительной производительности является неотделимым фрагментом процесса создания программного обеспечения. Основными критериями являются: частота процессора, оперативной памяти и жесткого диска, дополнительного специализированного оборудования. Для программного обеспечения, разработанного в соответствии с характеристиками, приведенными ниже. Поскольку программа требует сервера и клиента установки различаются для них.

На клиентских компьютерах не нужно устанавливать специальное программное обеспечение и компоненты, а стандартный браузер. Так как большинство машин под управлением Windows, разделенные и в качестве браузера, используя стандартный Internet Explorer, отображать параметры, перечисленные ниже для клиентского компьютера, в пределах параметров IE 8, рекомендованный Microsoft.

Серверная часть требует индивидуального подхода с точки зрения выбора аппаратных частей компьютера и зависит от количества подключенных клиентов, с учетом производственного потенциала запаса, чтобы иметь возможность подключения новых пользователей. Вся производственная мощность серверной части системы состоит из требований операционной системы, MS SQL Server. Разработанная система не привязана к конкретной операционной системе, и, следовательно, может быть установлен на любой, но рекомендуется использовать сетевые решения операционных систем, поскольку они специально предназначены для работы в сети.

4 Разработка БД и описание интерфейса приложения

4.1 Анализ предметной области

Название предметной области «Разработка ПО для медицинских учреждений на языке программирования C#».

ПО для медицинских учреждений – это программное обеспечение предназначенное для оптимизации работы государственных медицинских учреждений.

Предварительная модель базы данных представлены на рисунке 3:

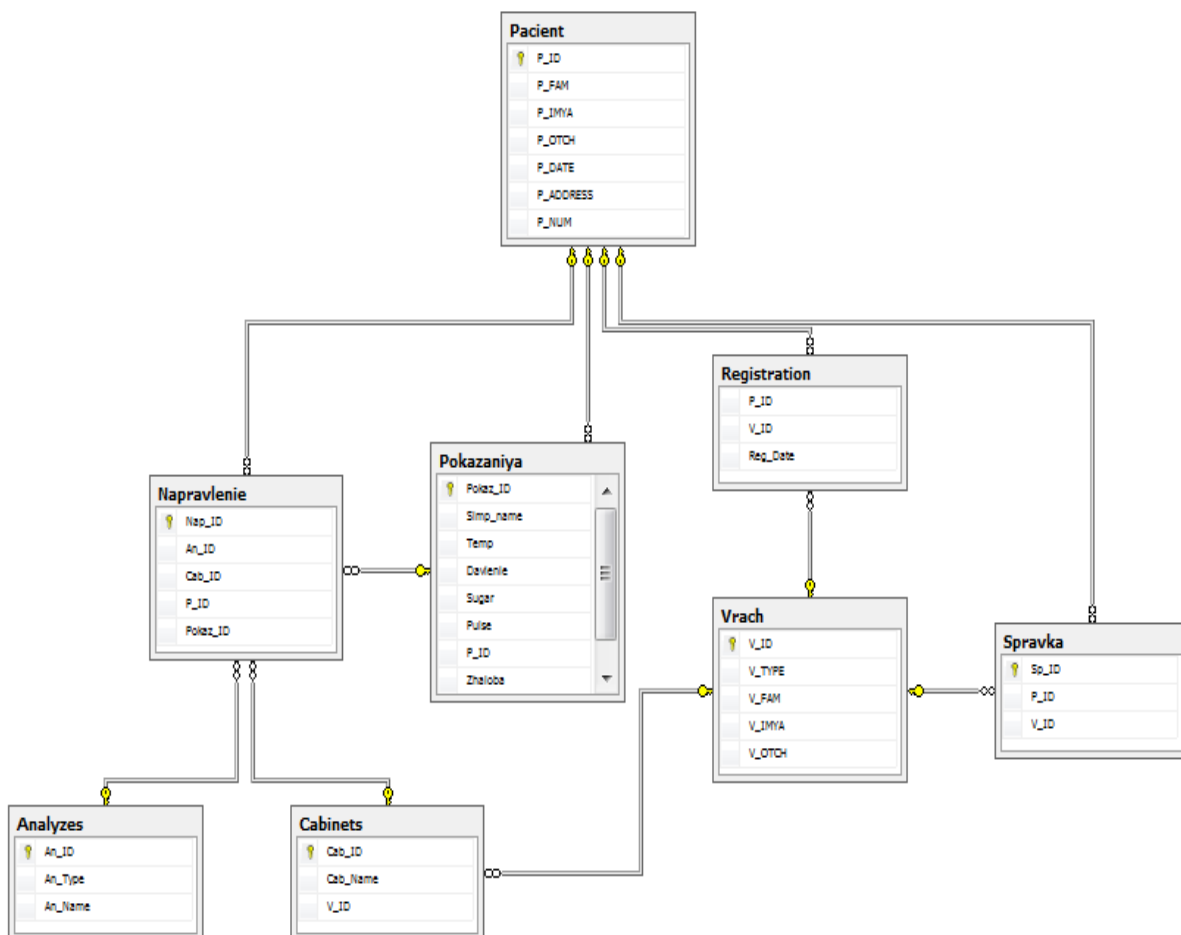


Рисунок 3 – Предварительная модель «ПО для поликлиники»

Т а б л и ц а 2 – Атрибуты каждой сущности

Имя сущности	Описание	Местонахождение экземпляров сущности
Расient	Пациенты поликлиники	Пациент приходит в поликлинику

Vrach	Рабочий персонал поликлиники	Врачи работают в поликлинике
Pokazaniya	Показания пациентов	Показания снимаются у пациентов, на их основе формируются направления
Analyzes	Анализы которые должен сдать пациент	Список анализов указывается в направлении
Cabinets	Врачебные кабинеты	В кабинетах сидят врачи, список кабинетов, который необходимо пройти пациенту указывается в направлении
Registration	Регистрация в базу пациентов	Пациент проходит процедуру регистрации, осуществляемую врачом
Spravka	Справка	Справку составляет врач, основываясь на первичных показаниях, а также на результатах анализов

Проанализируем контекстную диаграмму:

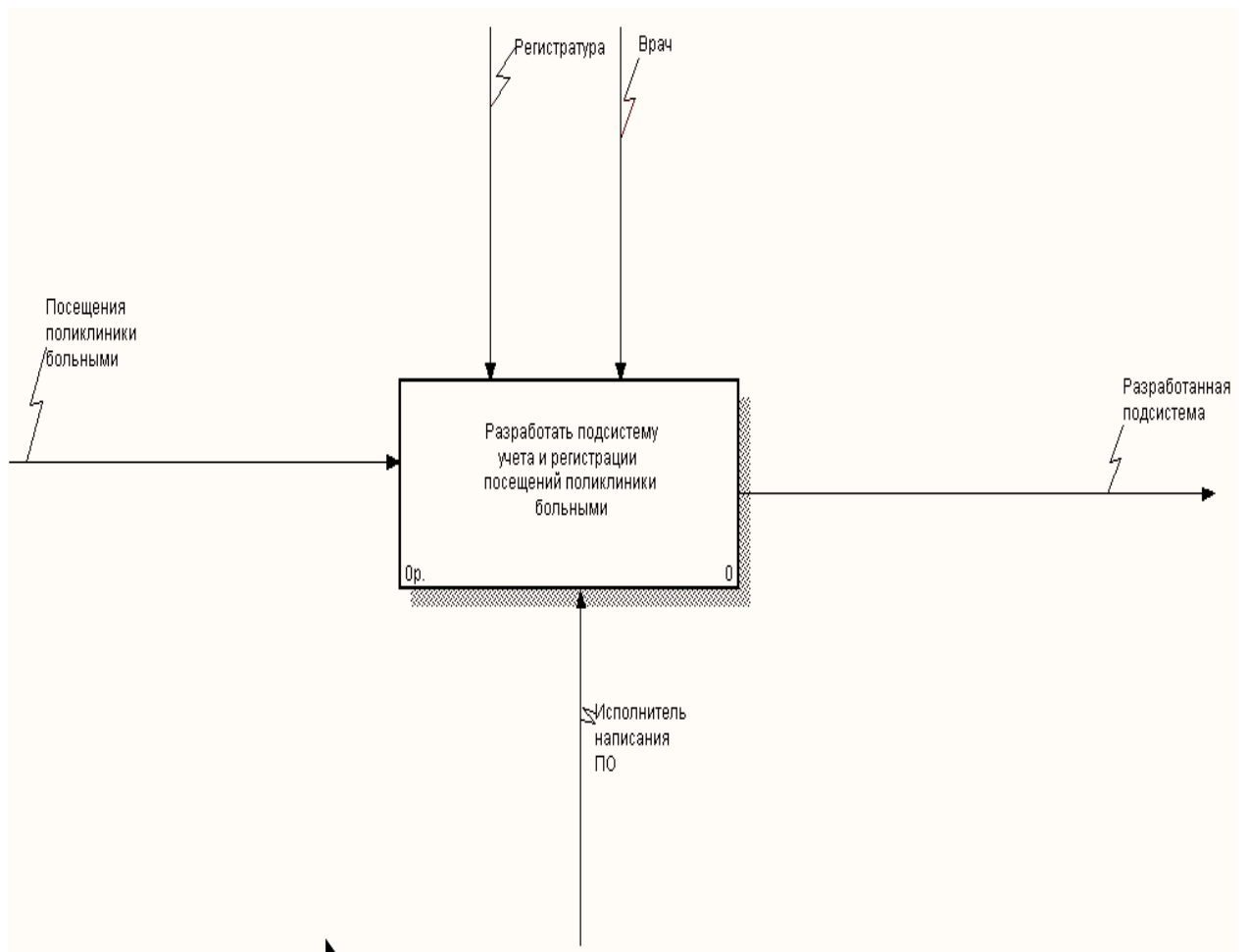


Рисунок 3.1 - Контекстная диаграмма модели учета посещений поликлиники

Входами функционального блока являются:

- 1) Посещения поликлиники больными – является инициатором

Выходы:

- 1) Разработанная подсистема

Механизмы:

- 1) Исполнитель написания ПО – осуществляет написание программного продукта

Контроль:

- регистратура – формирование начальных данных для дальнейшего формирования записей по посещению поликлиники.
- врач – выставление диагноза и рекомендаций по результатам обследования пациента

4.2 UML диаграммы

Unified Modeling Language (UML - Unified Modeling Language)

Это стандартный инструмент для создания документально ("чертежи") программного обеспечения. С UML, вы можете визуализировать, указать, построить, и процесс разработки программного обеспечения системы документов. UML предназначен для удовлетворения потребностей в моделировании любой системы из корпоративных информационных систем для распределенных веб-приложений, и даже в режиме реального времени встроенных систем. Это выразительный язык, который позволяет системе рассматривать все точки зрения, связанные с его развитием и последующего развертывания. Несмотря на обилие выразительных возможностей, язык легко понять и использовать.

Моделирование необходимо для понимания системы. Как правило, только с одной моделью, она никогда не бывает достаточно. Наоборот, для понимания практически любого нетривиального системы необходимо разработать большое количество взаимосвязанных моделей. Применительно к системам программного обеспечения, то это означает, что должен быть язык, который может быть использован для описания различных точек зрения представление архитектуры системы для своего цикла разработки.

На сегодняшний день существует 12 диаграмм UML 2.0:

- Диаграмма классов
- Диаграмма компонентов
- Диаграмма развёртывания
- Диаграмма прецедентов
- Диаграмма последовательности
- Диаграмма композитной/составной структуры
- Диаграмма пакетов
- Диаграмма объектов
- Диаграмма автомата
- Диаграмма вариантов использования
- Диаграмма обзора взаимодействия
- Диаграмма синхронизации

4.2.1 Диаграмма прецедентов

Диаграмма прецедентов применяется для демонстрации возможностей различных пользователей в зависимости от предоставленных прав. Диаграмма прецедентов представлена на рисунке 4.

Определение акторов:

– Администратор

– User

Варианты использования:

- смотрит информацию
- просмотр списка пациентов
- удаление таблиц
- обновление таблиц
- просмотр истории болезней
- заполнение таблиц
- добавить пациента
- заполнение амбулаторных карточек
- просмотр списка анализов

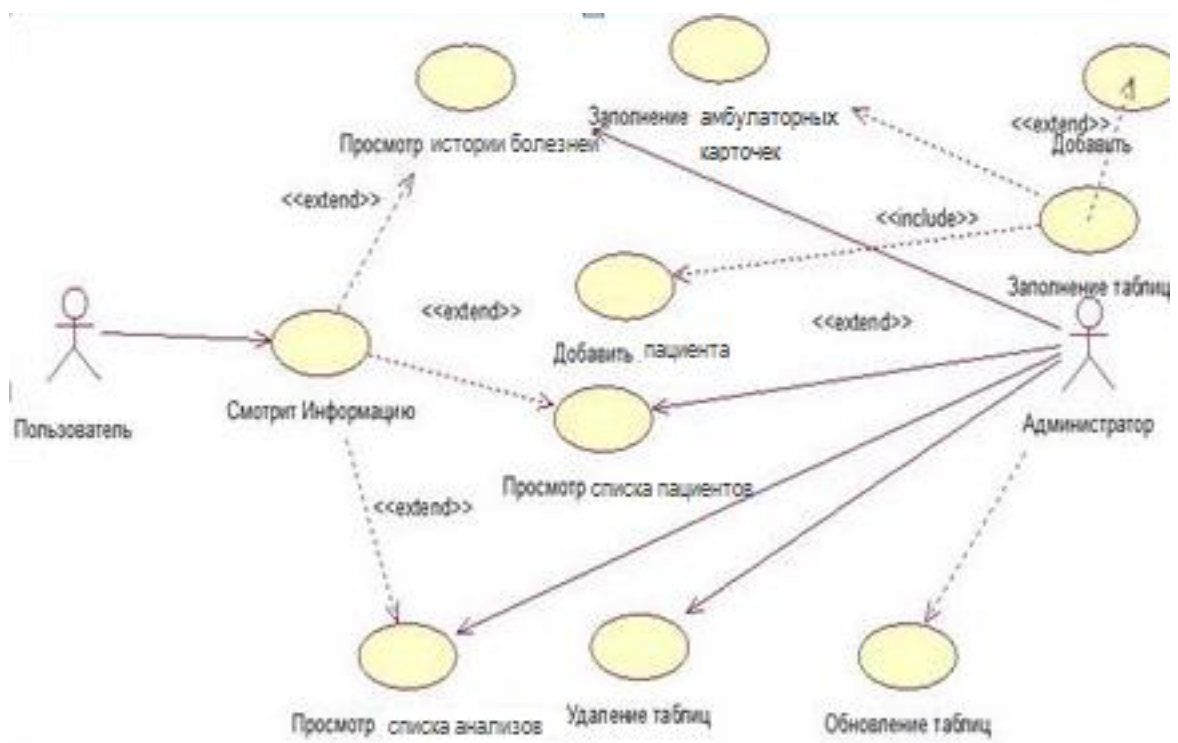


Рисунок 4 – Диаграмма прецедентов

Пояснение:

- Акторов – 2
- Количество ВИ – 10
- Включений – 1
- Расширений – 5

4.2.2 Диаграмма компонентов

Диаграмма компонент показывает набор компонентов и отношений между ними. Компонент - это физически сменная часть системы, которая имеет набор интерфейсов и обеспечивает их реализацию.

В этом случае исполняемый PO.exe применяется для добавления информации или к данным PO.mdf базы данных. Но звонки не напрямую, а через SQLServer. Таким образом, SQLServer представляет собой интерфейс между программой и базой данных, обеспечивающей их взаимодействие. Диаграмма компонент представлена на рисунке 5.

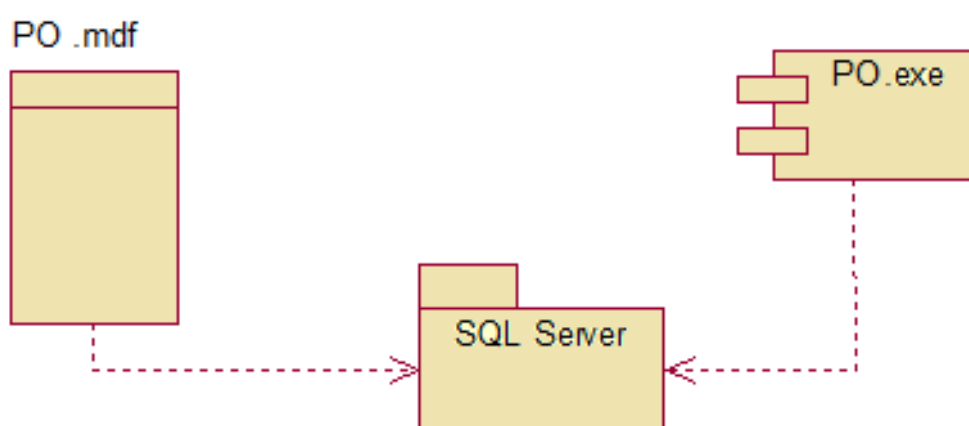


Рисунок 5 – Диаграмма компонент

4.2.3 Диаграмма классов

Диаграмма классов в UML является частным случаем ER диаграмм. ER диаграммы используются для логического проектирования баз данных. Их главное отличие: На диаграмме ER фокусируется на структуру данных, и на диаграмме классов - поведение классов.

4.2.4 Алгоритм работы программы

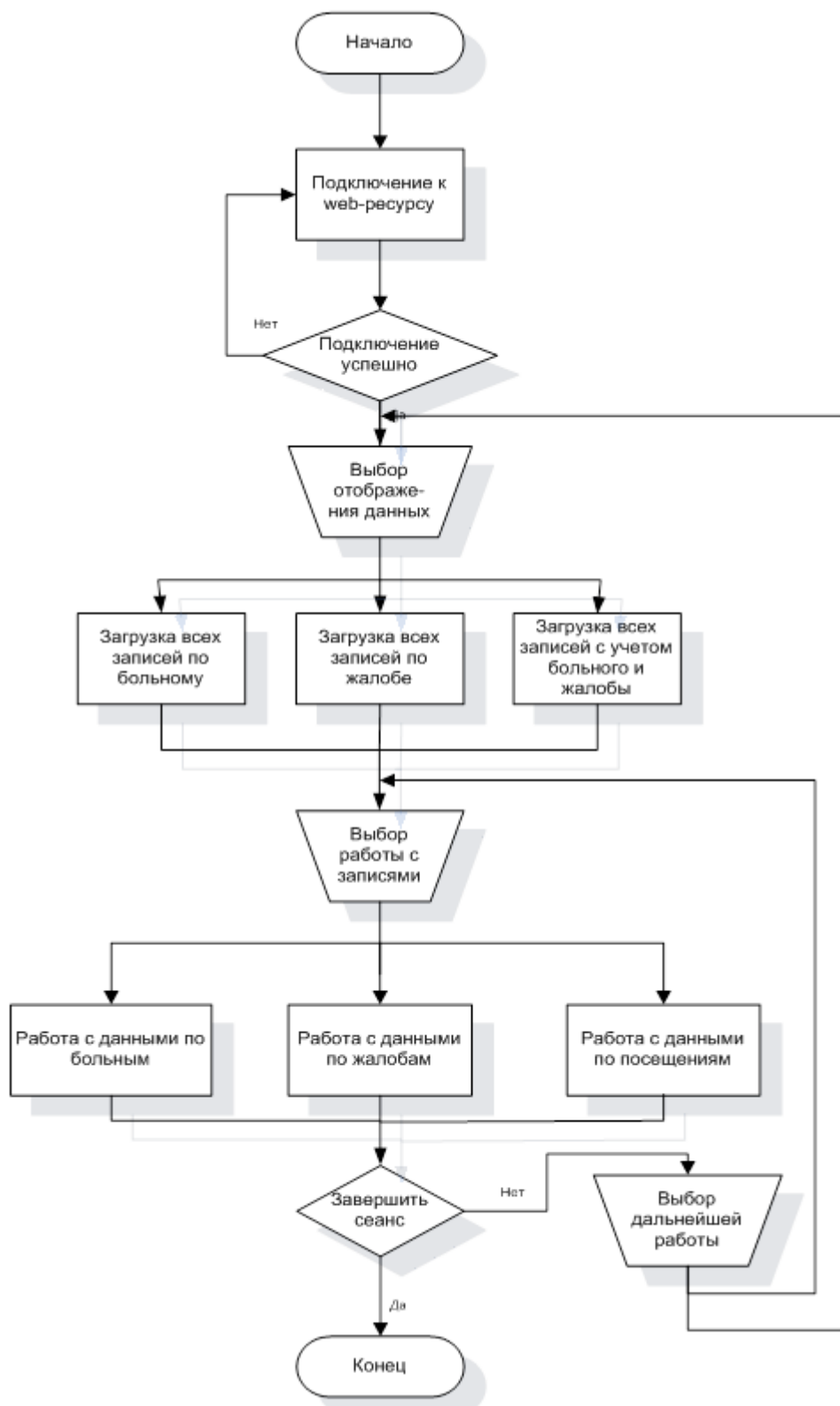


Рисунок 6 - Обобщенный алгоритм работы клиента.

4.3 Проектирование БД

4.3.1 Концептуальное проектирование

С учетом характера рассматриваемой системы при анализе предметной области, выбрать таблицы и связи между ними представлены в предварительном ER - диаграммы.

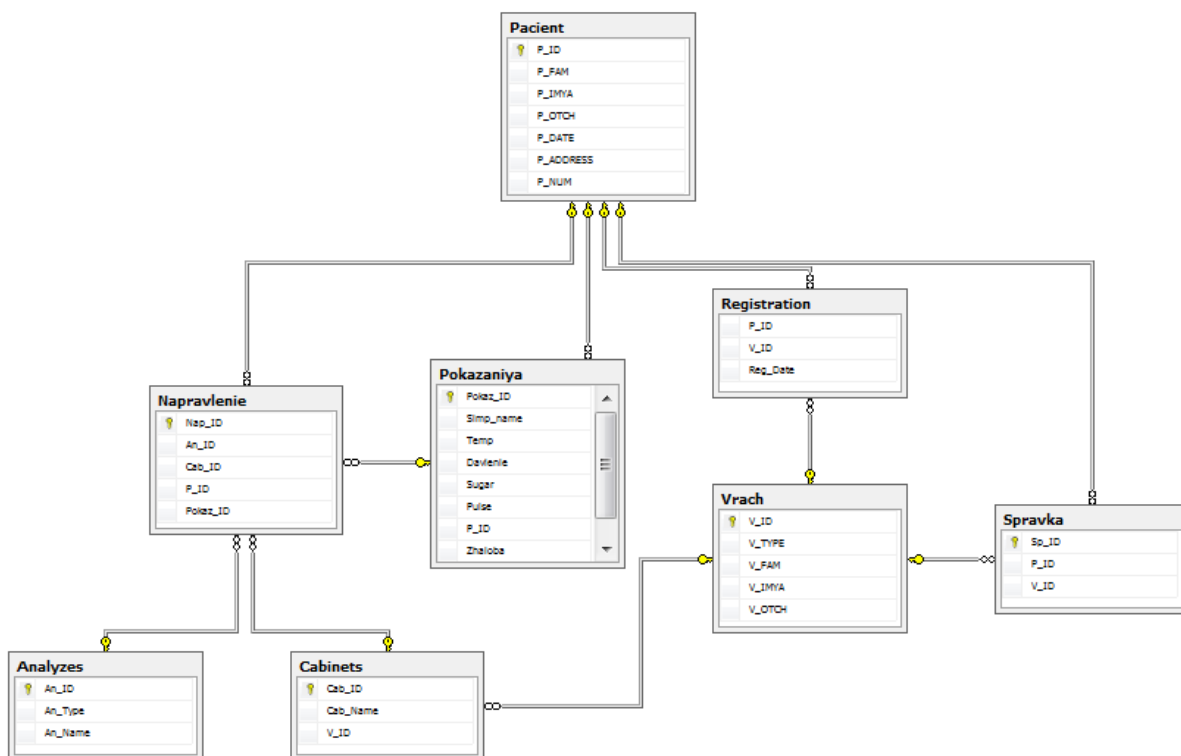


Рисунок 6 – Предварительная ER – диаграмма

На основе предварительной модели будут формироваться бизнес-правила. Затем определяют отношения между субъектами, в основном, на основе описания операций. Точнее отношения сущностей будут базироваться на основе бизнес-правил, которые основаны на подробном описании операций.

4.3.2 Разработка и построение подробной ER – диаграммы на основании бизнес-правил.

Бизнес – правило 1

У одного у одного пациента может быть одно направление, поэтому связь - «один-к-одному», и наоборот, одно направление может выдано только одному пациенту, поэтому связь - «один-к-одному» (рисунок 7).

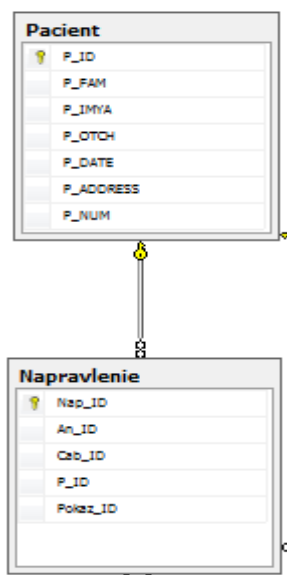


Рисунок 7 – Сегмент ER-модели для бизнес-правила 1

Бизнес-правило 2

У одного пациента одни показания, показания снимаются у одного пациента, следовательно, эти сущности объединяет связь «один-к-одному» (рисунок 8).

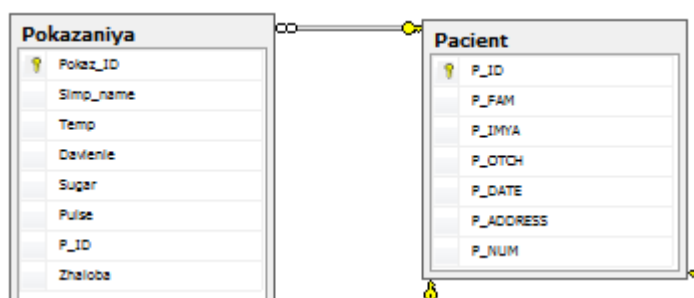


Рисунок 8 - Сегмент ER-модели для бизнес-правила 2

Бизнес – правило 3

Пациент регистрируется один раз, следовательно, эти сущности объединяет связь «один-к-одному» (рисунок 9).

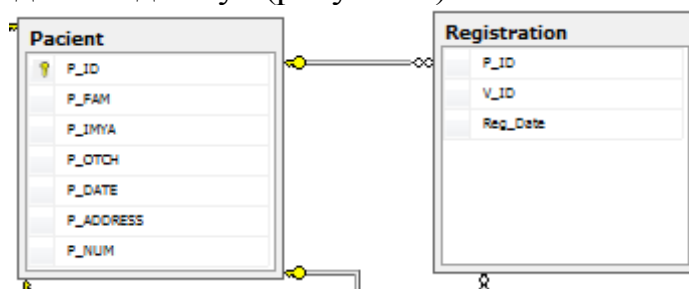


Рисунок 9 - Сегмент ER-модели для бизнес-правила 3

Бизнес – правило 4

Регистрацию одного пациента проводит один врач, следовательно это сущности объединяет связь «один-к-одному» (рисунок 10).

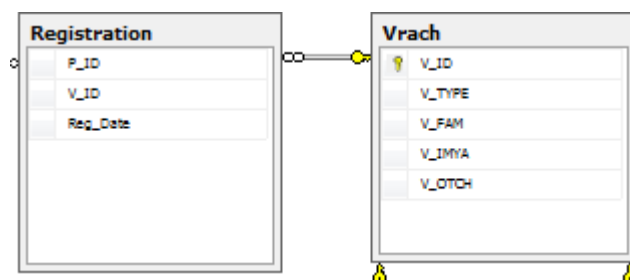


Рисунок 10 - Сегмент ER-модели для бизнес-правила 4

Бизнес – правило 5

Врач выписывает одну справку, следовательно, эти сущности объединяет связь «один-к-одному» (рисунок 11).

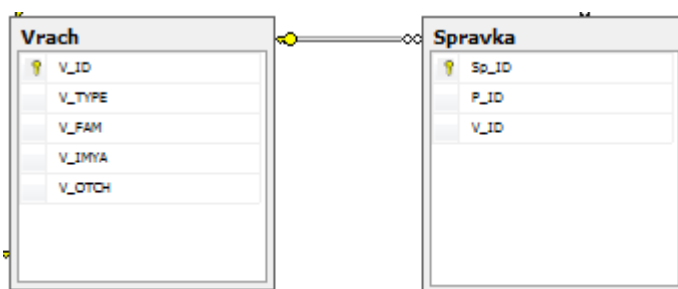


Рисунок 11 - Сегмент ER-модели для бизнес-правила 5

Бизнес-правило 6

Один врач одновременно может сидеть в одном кабинете, следовательно, эти сущности объединяет связь «один-к-одному» (рисунок 12).

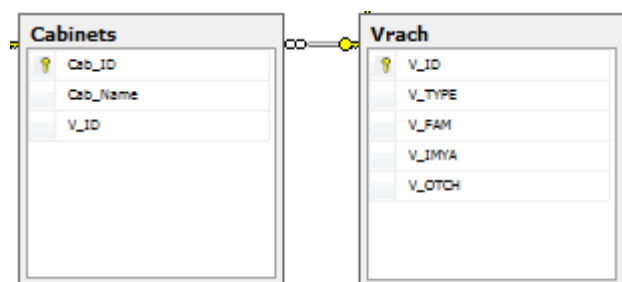


Рисунок 12- Сегмент ER-модели для бизнес-правила 6

Бизнес-правило 7

В направлении могут указываться несколько кабинетов, несколько кабинетов указываются в одном направлении, значит связь - «один-ко-многим»(рисунок 13).

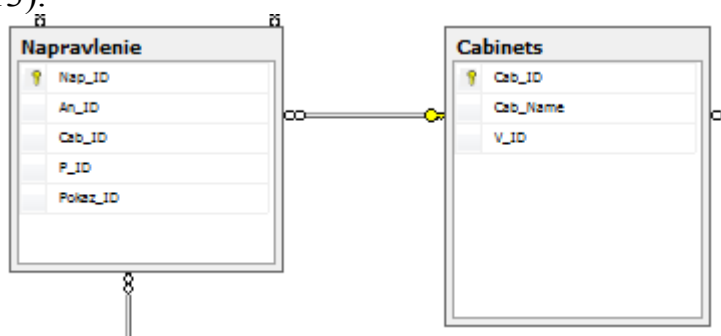


Рисунок 13- Сегмент ER-модели для бизнес-правила 7

Бизнес – правило 8

В направлении могут указываться несколько видов анализов, несколько видов анализов указываются в одном направлении, значит связь - «один-ко-многим» (рисунок 14).

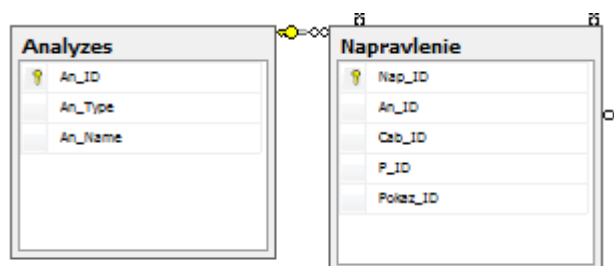


Рисунок 14 - Сегмент ER-модели для бизнес-правила 8

На рисунке 15 представлена уточненная ER – диаграмма концептуальной модели «ПО для поликлиник».

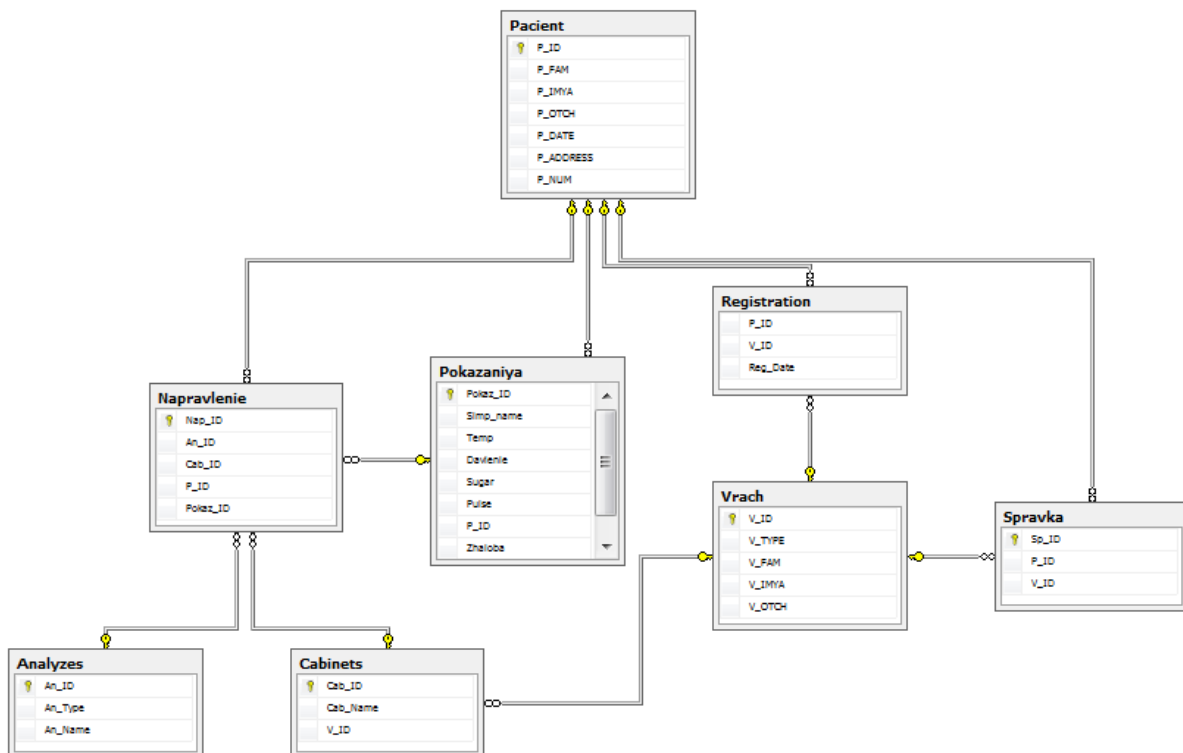
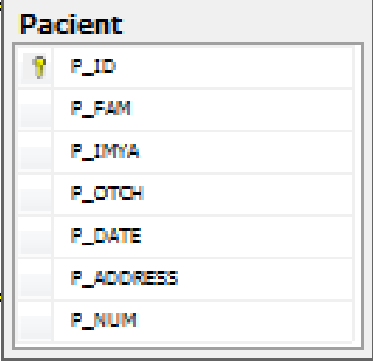
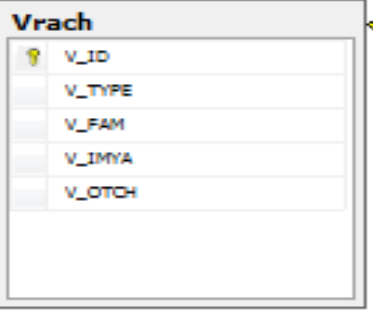


Рисунок 15 – уточненная ER-диаграмма концептуальной модели «ПО для поликлиники»

4.3.3 Определение атрибутов каждой сущности

Т а б л и ц а 3 – Атрибуты каждой сущности

Сегмент ER - модели	Описание
	<p>Таблица Pacient содержит следующие данные: P_ID – уникальный код пациента; P_FAM – фамилия пациента P_IMYA – имя пациента; P_OTCH– отчество пациента; P_DATE – дата рождения пациента; P_ADDRESS – адрес пациента; P_NUM – номер телефона пациента;</p>
	<p>Таблица Vrach содержит следующие данные: V_ID – уникальный код врача; V_Type – тип врача; V_FAM – фамилия врача; V_IMYA – имя врача; V_OTCH– отчество врача;</p>

<table border="1"> <thead> <tr> <th colspan="2">Registration</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>P_ID</td> </tr> <tr> <td><input type="checkbox"/></td> <td>V_ID</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Reg_Date</td> </tr> </tbody> </table>	Registration		<input type="checkbox"/>	P_ID	<input type="checkbox"/>	V_ID	<input type="checkbox"/>	Reg_Date	<p>Таблица Registration содержит следующие данные: P_ID – уникальный код пациента; V_ID – уникальный код врача; Reg_Date – дата регистрации пациента;</p>										
Registration																			
<input type="checkbox"/>	P_ID																		
<input type="checkbox"/>	V_ID																		
<input type="checkbox"/>	Reg_Date																		
<table border="1"> <thead> <tr> <th colspan="2">Spravka</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>Sp_ID</td> </tr> <tr> <td><input type="checkbox"/></td> <td>P_ID</td> </tr> <tr> <td><input type="checkbox"/></td> <td>V_ID</td> </tr> </tbody> </table>	Spravka		<input checked="" type="checkbox"/>	Sp_ID	<input type="checkbox"/>	P_ID	<input type="checkbox"/>	V_ID	<p>Таблица Spravka содержит следующие данные: Sp_ID – уникальный код справки; P_ID – уникальный код пациента; V_ID – уникальный код врача;</p>										
Spravka																			
<input checked="" type="checkbox"/>	Sp_ID																		
<input type="checkbox"/>	P_ID																		
<input type="checkbox"/>	V_ID																		
<table border="1"> <thead> <tr> <th colspan="2">Cabinets</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>Cab_ID</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Cab_Name</td> </tr> <tr> <td><input type="checkbox"/></td> <td>V_ID</td> </tr> </tbody> </table>	Cabinets		<input checked="" type="checkbox"/>	Cab_ID	<input type="checkbox"/>	Cab_Name	<input type="checkbox"/>	V_ID	<p>Таблица Cabinets содержит следующие данные: Cab_ID – номер кабинета; Cab_Name – название кабинета; V_ID – уникальный код врача;</p>										
Cabinets																			
<input checked="" type="checkbox"/>	Cab_ID																		
<input type="checkbox"/>	Cab_Name																		
<input type="checkbox"/>	V_ID																		
<table border="1"> <thead> <tr> <th colspan="2">Pokazaniya</th> </tr> </thead> <tbody> <tr> <td><input checked="" type="checkbox"/></td> <td>Pokaz_ID</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Simp_name</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Temp</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Davlenie</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Sugar</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Pulse</td> </tr> <tr> <td><input type="checkbox"/></td> <td>P_ID</td> </tr> <tr> <td><input type="checkbox"/></td> <td>Zhaloba</td> </tr> </tbody> </table>	Pokazaniya		<input checked="" type="checkbox"/>	Pokaz_ID	<input type="checkbox"/>	Simp_name	<input type="checkbox"/>	Temp	<input type="checkbox"/>	Davlenie	<input type="checkbox"/>	Sugar	<input type="checkbox"/>	Pulse	<input type="checkbox"/>	P_ID	<input type="checkbox"/>	Zhaloba	<p>Таблица Pokazaniya содержит следующие данные: Pokaz_ID – уникальный код; Simp_Name – симптомы; Temp – температура; Davlenie – давление пациента; Sugar – уровень сахара в крови; Pulse – пульс пациента; P_ID – уникальный код пациента; Zhaloba – жалобы пациента;</p>
Pokazaniya																			
<input checked="" type="checkbox"/>	Pokaz_ID																		
<input type="checkbox"/>	Simp_name																		
<input type="checkbox"/>	Temp																		
<input type="checkbox"/>	Davlenie																		
<input type="checkbox"/>	Sugar																		
<input type="checkbox"/>	Pulse																		
<input type="checkbox"/>	P_ID																		
<input type="checkbox"/>	Zhaloba																		

<p>Napravlenie</p> <ul style="list-style-type: none"> ⚡ Nap_ID An_ID Cab_ID P_ID Pokaz_ID 		<p>Таблица Napravlenie содержит следующие данные: Nap_ID – номер направления; An_ID – уникальный код анализа; Cab_ID – номер кабинета; P_ID – уникальный код пациента; Pokaz_ID – уникальный код;</p>
<p>Analyzes</p> <ul style="list-style-type: none"> ⚡ An_ID An_Type An_Name 		<p>Таблица Analyzaes содержит следующие данные: An_ID – уникальный код анализа; An_Type – вид анализа; An_Name – название анализа.</p>

4.3.4 Расчет места для хранения БД

На данном этапе необходимо знать, сколько места будет создан для хранения базы данных. Объем внешней памяти, необходимой для функционирования системы состоит из двух компонентов: память, занятую модулями базы данных (основные инструменты, аксессуары), и памяти, зарезервированной для данных (M_d). Наиболее важным является, как правило, M_d . Объем памяти, занимаемая программными модулями пользователя, как правило, мал по сравнению с объемом самих данных, поэтому он не может быть принят во внимание. Проект рассчитан в предположении, что максимальный объем памяти, занимаемый базой данных.

Объем памяти, занимаемая программными модулями пользователя, как правило, мал по сравнению с объемом самих данных, поэтому он не может быть принят во внимание. Требуемый объем памяти определяется на основе интенсивности запросов и анализа объема полученных данных.

4.4 Логическое проектирование

Задачей логического проектирования является создание таблицы реляционной структуры на языке DDL.

4.4.1 Меры по обеспечению безопасности

Одним из наиболее важных моментов базы данных является предоставление права доступа к нему, потому что они нуждаются в защите от несанкционированного доступа и защиты доступа. Для безотказной стратегии резервного копирования разрабатывается. Для защиты от несанкционированного доступа к любым пользовательским данным, доступным только в соответствии с их правами доступа.

4.4.2 Создание пользователей, логинов и задание им паролей

Одним из наиболее важных компонентов проекта базы данных является разработка инструментов для защиты базы данных. Защита данных имеет два аспекта: защиту от сбоев и защиты от несанкционированного доступа. Для безотказной стратегии резервного копирования разрабатывается. Для защиты от несанкционированного доступа к любым пользовательским данным, доступным только в соответствии с их правами доступа. Для базы данных на основе анализа предметной области создаются

4 учетные записи: администратор, пользователь, редактор таблиц, редактор данных таблиц. Администратору даются все полномочия на нашу БД, а пользователю предоставляется возможность только просматривать информацию.

Создание пользователей и выдача им прав:

```
CREATE USER Врач IDENTIFIED BY 123456 DEFAULT
TABLESPACE USERS TEMPORARY TABLESPACE TEMP;
CREATE USER Специалист IDENTIFIED BY 123456 DEFAULT
TABLESPACE USERS TEMPORARY TABLESPACE TEMP;
CREATE USER Лаборатория IDENTIFIED BY 123456 DEFAULT
TABLESPACE USERS TEMPORARY TABLESPACE TEMP;
```

```
GRANT CREATE SESSION TO Врач;
GRANT CREATE SESSION TO Специалист;
GRANT CREATE SESSION TO Лаборатория;
```

```
GRANT dba TO Врач;
```

4.4.3 Создание таблиц

Создание таблицы Patient:

```
Create table Patient (
```

```
P_ID bigint not null Primary key,
```

```
P_FAM char(20) NOT NULL,
```



```
P_IMYA char(10),  
P_OTCH char(15),  
P_DATE datetime,  
P_ADDRESS char(25),  
P_NUM char(10)  
);
```

*Создание таблицы **Vrach**:*

```
Create table Vrach (  
V_ID bigint not null Primary key,  
V_TYPE char(50),  
V_FAM char(20) NOT NULL,  
V_IMYA char(10),  
V_OTCH char(15)  
);
```

*Создание таблицы **Analyzes**:*

```
Create table Analyzes (  
An_ID bigint not null Primary key,  
An_Type varchar(100),  
An_Name varchar(50)/*название анализа*/  
);
```

*Создание таблицы **Cabinets**:*

```
Create table Cabinets(  

```

Cab_ID bigint not null primary key,

Cab_Name char(30),

V_ID bigint not null foreign key references Vrach(V_ID));

Создание таблицы Pokazaniya:

Create table Pokazaniya (

Pokaz_ID bigint not null Primary key,

Simp_name char(50),/*название симптома*/

Temp char(10) NOT NULL,/*температура*/

Davlenie char(10) NOT NULL,

Sugar char(15) NOT NULL, /*уровень сахара*/

Pulse char(15), /*пульс*/

Zhaloba varchar(200) ,

P_ID bigint not null foreign key references Pacient(P_ID)

);

Создание таблицы Registration:

Create table Registration

(

P_ID bigint not null foreign key references Pacient(P_ID),

V_ID bigint not null foreign key references Vrach(V_ID),

Reg_Date datetime);

Создание таблицы Spravka:

```
Create table Spravka (  
Sp_ID bigint not null Primary key,  
P_ID bigint not null foreign key references Pacient(P_ID),  
V_ID bigint not null foreign key references Vrach(V_ID),  
);
```

*Создание таблицы **Napravlenie**:*

```
Create table Napravlenie --направление на анализы  
(  
Nap_ID bigint not null Primary key,  
An_ID bigint not null foreign key references Analyzes(An_ID),  
Cab_ID bigint not null foreign key references Cabinets(Cab_ID),/*номер кабинета*/  
P_ID bigint not null foreign key references Pacient(P_ID),  
Pokaz_ID bigint not null foreign key references Pokazaniya(Pokaz_ID),  
);
```

4.5 Описание интерфейса приложения. Работа с приложением

Интерфейс данного приложения очень удобен для использования. С приложение могут работать три вида пользователей: врач, специалист и лаборатория. В зависимости от выбора пользователя, .

После запуска приложения открывается первая форма - форма гостя. На данном этапе гость просматривает данные таблиц (рисунок 4.1). Зайдя на формы таблиц, кнопка «Update» будет отключена (рисунок 4.3). Чтобы изменять данные надо зайти в форму «Login»и пройти аутентификацию пользователя.

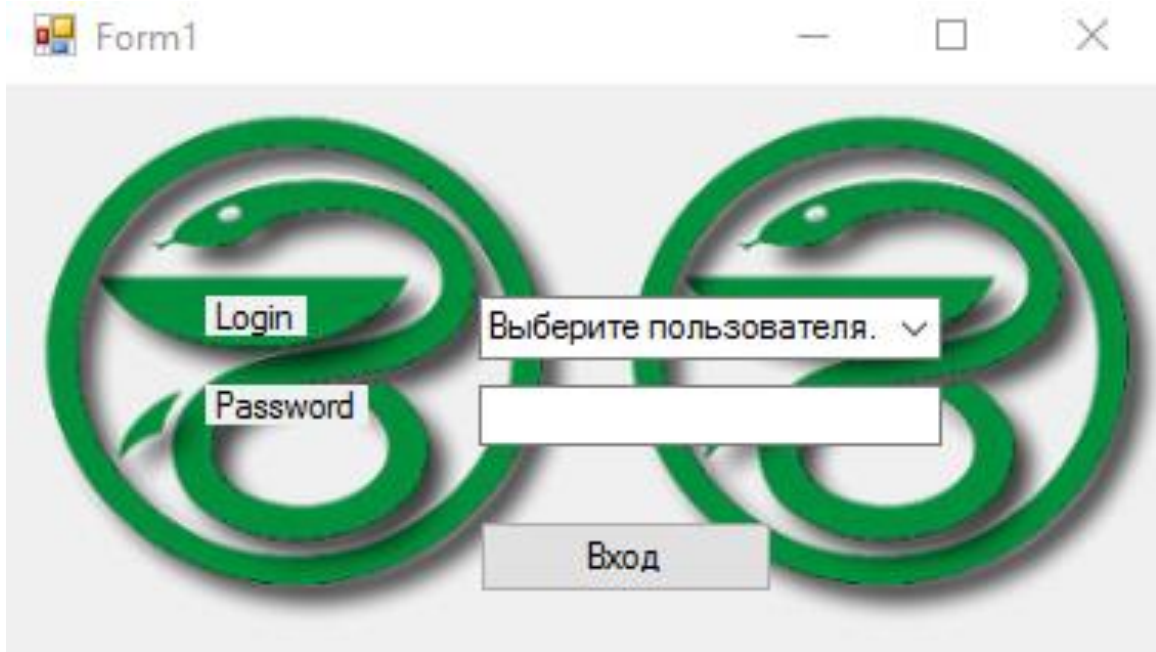


Рисунок 4.1 – Главная форма

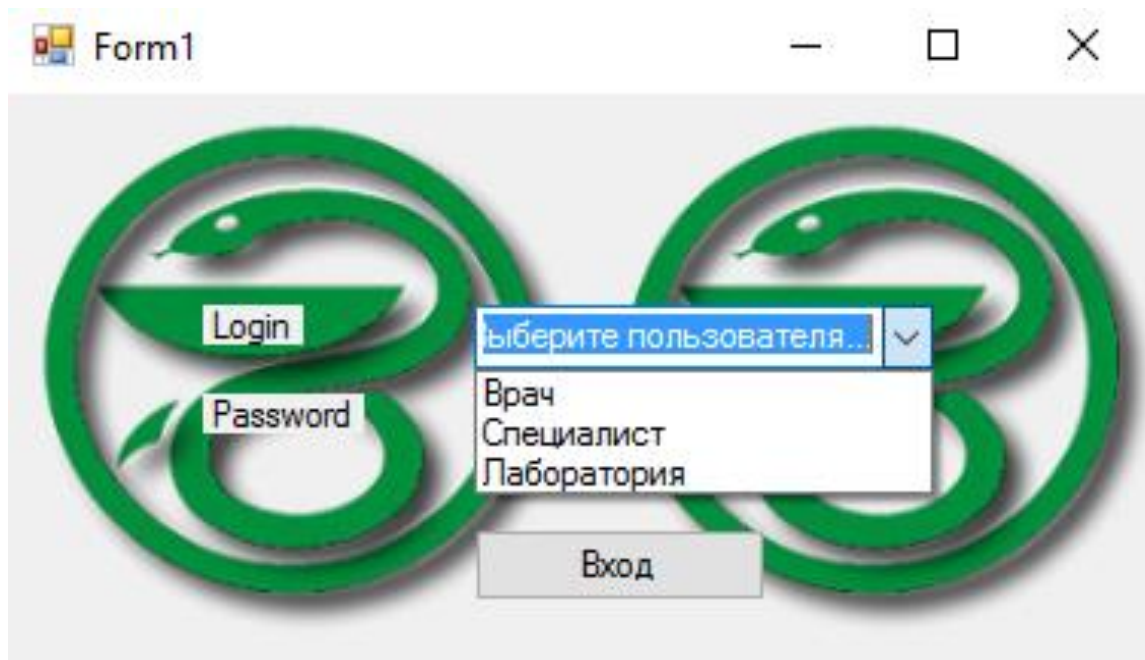


Рисунок 4.2 – Выбор пользователя

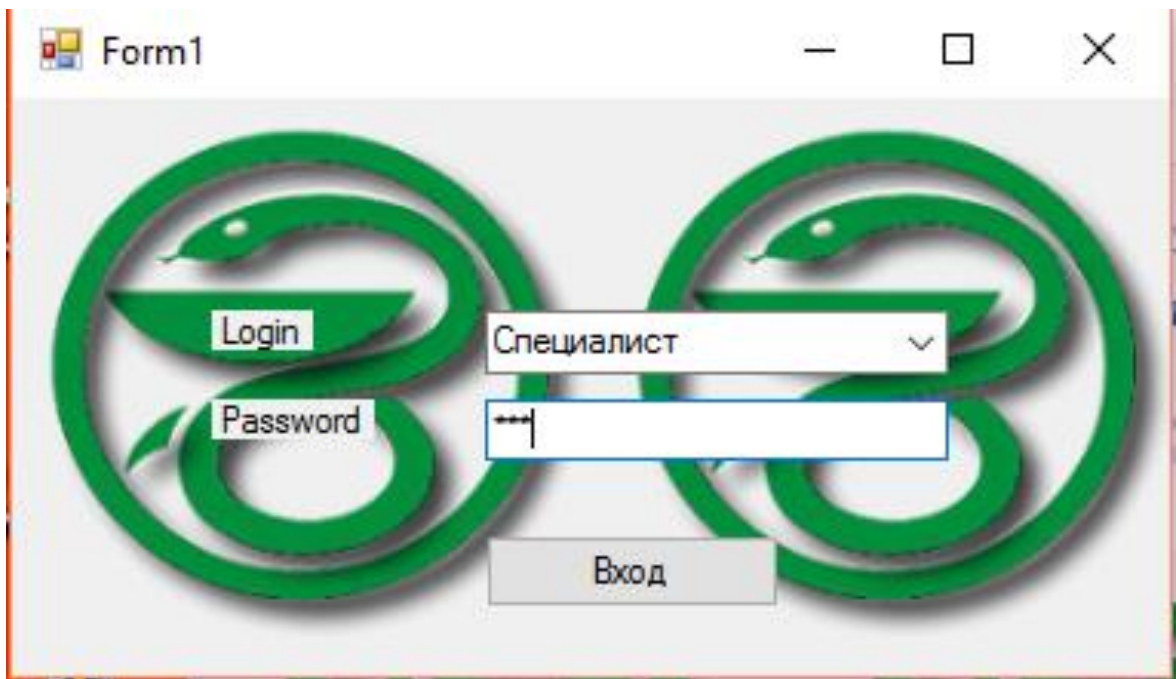


Рисунок 4.3 – Вход от имени специалиста

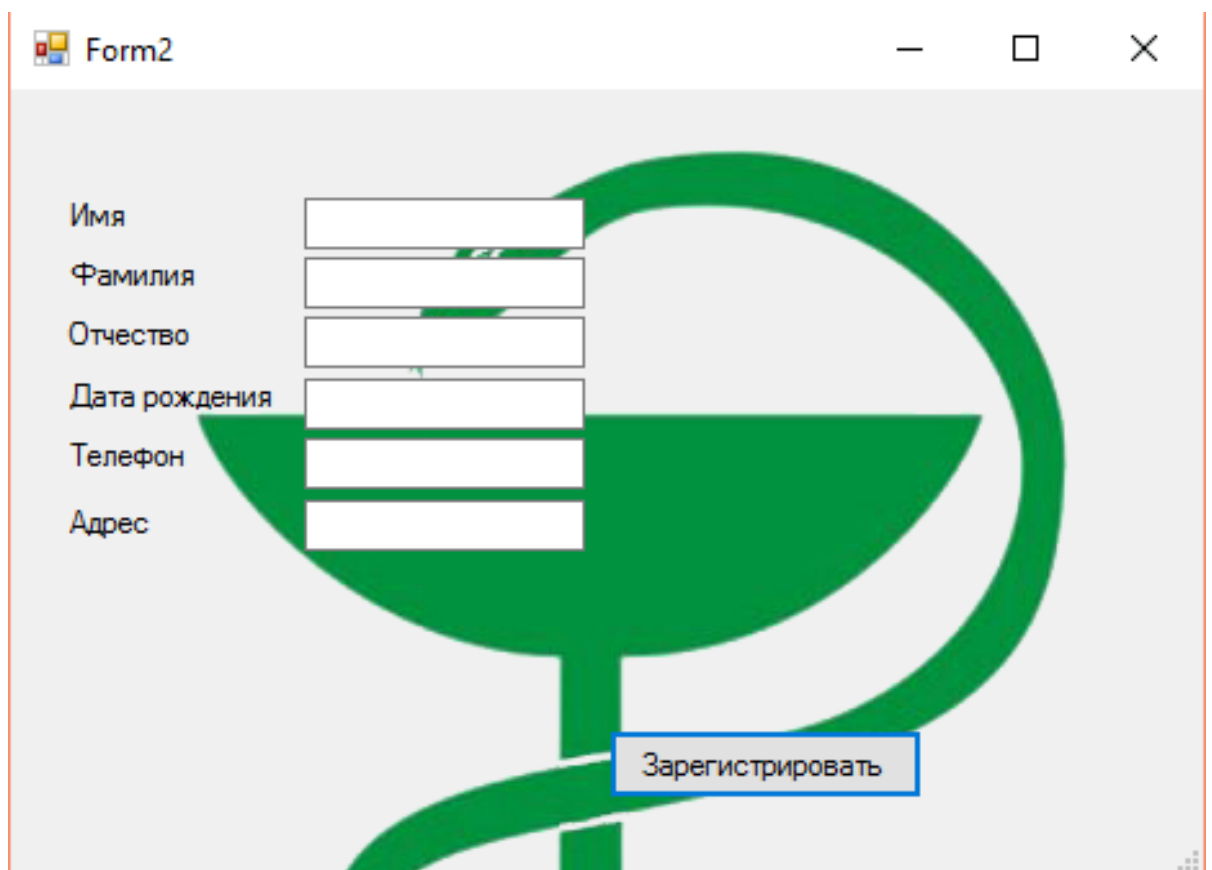


Рисунок 4.4 – Форма регистрации пациента

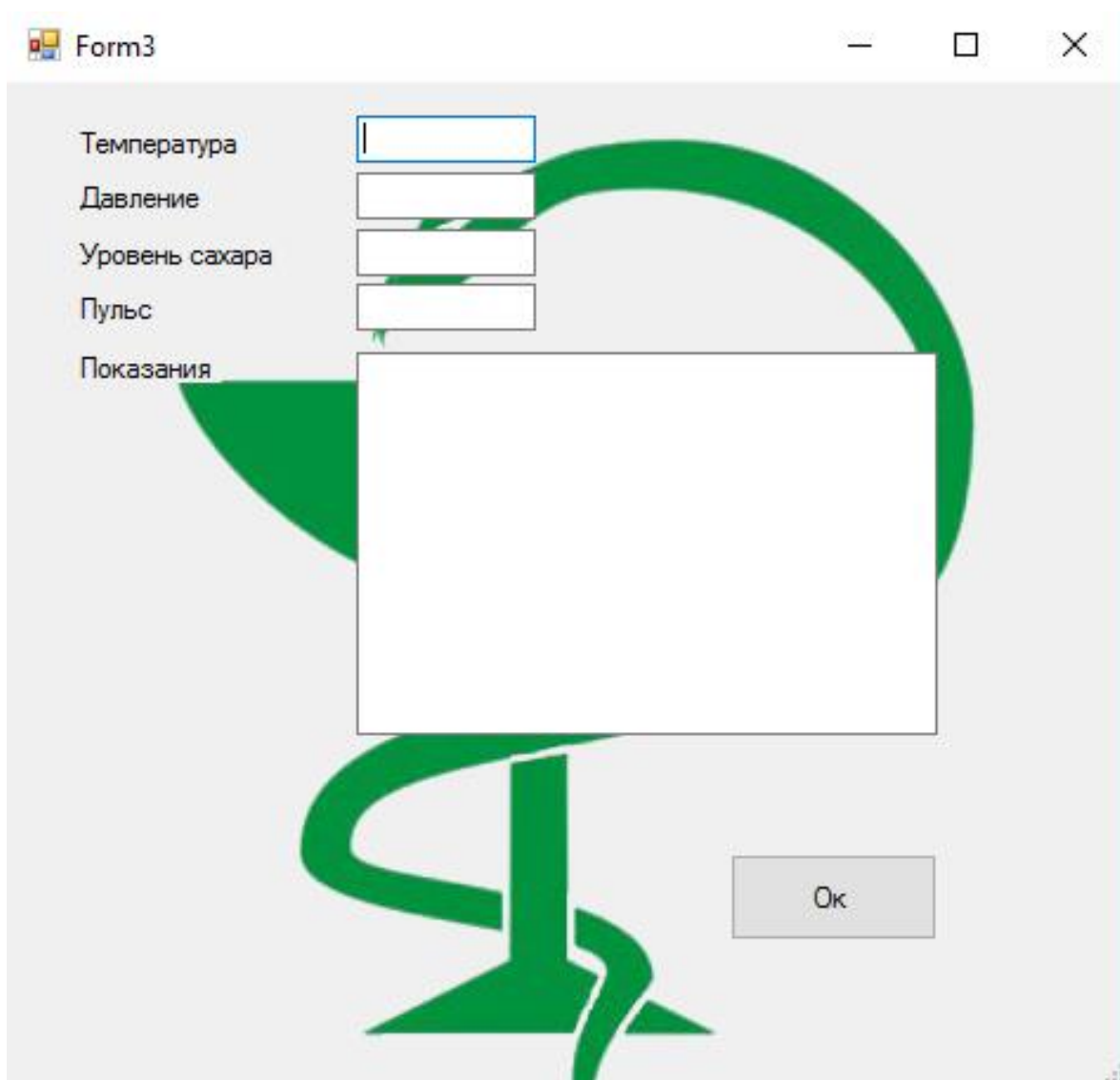


Рисунок 4.5 – Форма регистрации пациента

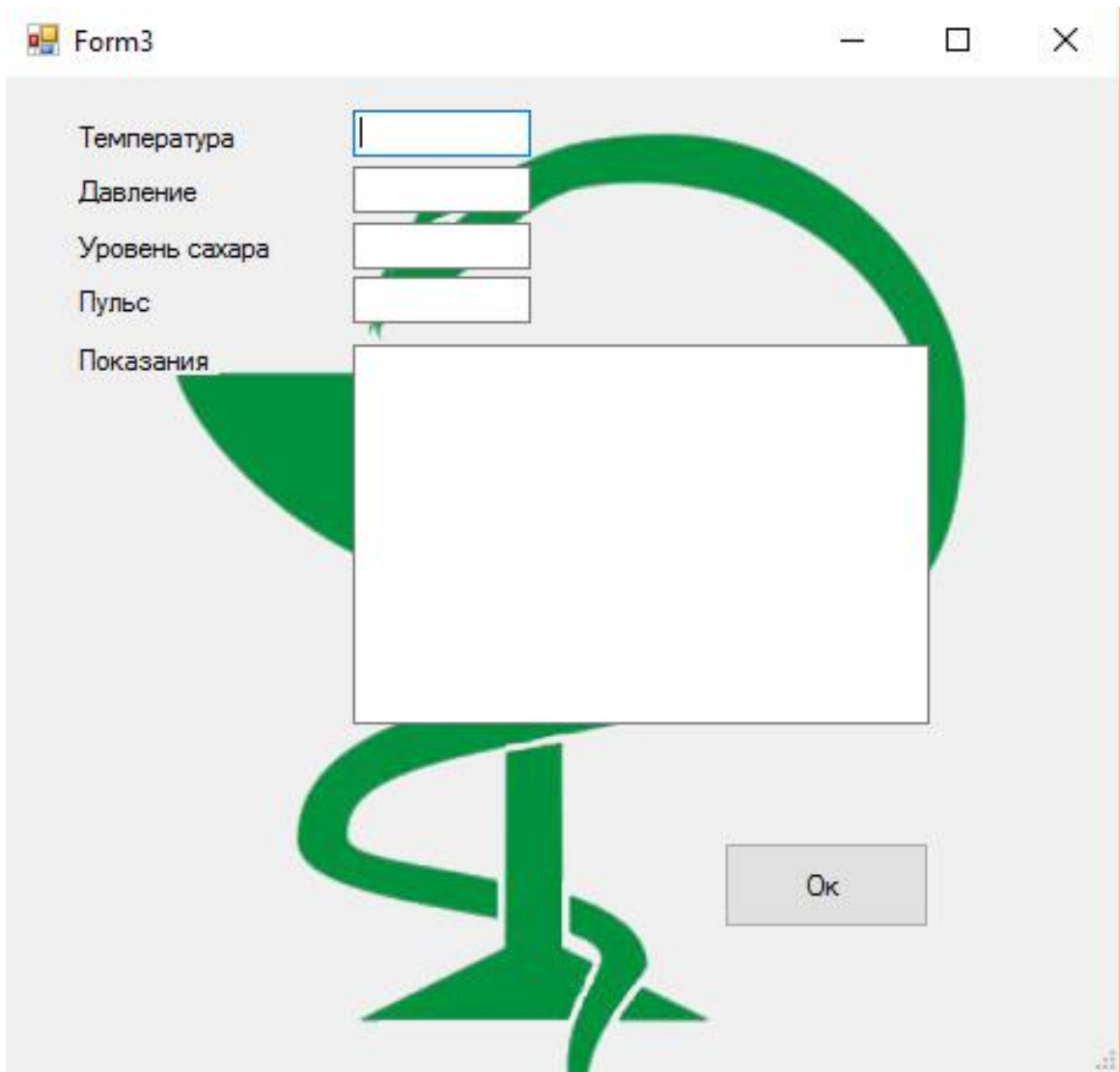


Рисунок 4.6 – Форма снятия показаний у пациента

Form6

Направление №

ФИО

Список анализов

Кровь	Каб.№3
Микрореакция	Каб.№55
Флюорографии	Каб.№78
Анализ мочи	Каб.№76

Место для печати

Распечатать

Рисунок 4.7 – Формирование направления

An_ID	An_Type	An_Name
2	Исследование крови и костного мозга	Развернутый анализ крови
3	Исследование крови и костного мозга	Общий анализ крови с тромбоцитами
4	Исследование крови и костного мозга	Общий анализ крови
5	Исследование крови и костного мозга	Обнаружение клеток красной волчанки
6	Исследование мочи	Общий анализ мочи
7	Исследование мочи	Анализ мочи по Нечипоренко
8	Исследование мочи	Определение глюкозы в моче
9	Исследование мочи	Анализ мочи по Зимницкому
10	Исследование мочи	Определение билирубина в моче
11	Исследование мочи	Суточный белок в моче(протеинурия)
12	Исследование мочи	Активные лейкоциты в моче
13	Исследование мочи	Определение оксалатов

Рисунок 4.8 – Список анализов

Справка

Файл Просмотр истории Выход

СПРАВКА №

Дата и время приема

Пациент № ФИО Врач

Дата рождения Адрес

Жалобы при обращении :

Со слов пациента :

Status praesens(подробно):

Предварительный диагноз :

Лечебные рекомендации:

Дополнительно:

Рисунок 4.9 – Шаблон справок

5 Экономическое обоснование проекта

5.1 Описание работы и обоснование необходимости

Тема данного дипломного проекта – «Разработка программного обеспечения для медицинских учреждений с использованием технологии C#».

Цель проекта – разработать программный продукт, который способствовал бы оптимизации работы медицинских учреждений.

Медицина является одним из актуальных наук с огромным количеством материалов, в ходе разработки данной системы взят и объединен в одно, наиболее важный материал из разных книг и пособий, что позволяет пациенту изучить самые необходимые аспекты по данной науке.

В этом разделе представлен обзор осуществления экономической составляющей этого продукта, который отражает время, трудовые и финансовые затраты проекта.

5.2 Трудовые ресурсы, используемые в работе.

В соответствии со спецификой и сложностью функций производится комплектование командой специалистов-художников, участвующих в разработке программного обеспечения, определение образования, профессии, квалификации и названия представлены в таблице 5.1.

Таблица 5.1 – Сотрудники и их заработная плата

Специалист-исполнитель	Количество человек	Зарботная плата в месяц, тенге
Программист	1	150000
Итого	1	150000

5.3 Оборудование, используемое в работе

Оборудование, используемое при разработке программного продукта представлено в таблице 5.2.

Таблица 5.2 – Перечень оборудования, необходимого для разработки

Наименование изделий	Характеристика	Количество единиц	Цена за единицу, тенге	Общая сумма, тенге
Ноутбук	Intel Core i7-5500U 6Gb DDR3 GeForce GT 940M 2GB HDD 1000Gb	1	75000	75000

	15.6" LED LCD (16:9)			
Многофункциональное устройство	Samsung SL-M2070	1	30000	30000
Итого				105000

Цены на оборудование представлены без учета НДС.

5.4 Программное обеспечение, используемое в работе

При разработке программы для медицинских учреждений было использовано следующее программное обеспечение:

- Windows 7 – операционная система;
- MS SQL Server 2010 – Средство Управления Базой Данных;
- MS Visual Studio 2013 – Среда программирования.

Программное обеспечение, использованное при разработке информационной системы, представлено в таблице 5.3.

Таблица 5.3 – перечень программного обеспечения, необходимого для разработки информационной системы

Программное обеспечение	Стоимость, тенге
Windows 7 максимальная	15000
MS SQL Server 2010	30000
MS Visual Studio 2013	40000
Итого	85000

Цены на ПО приведены без учета НДС.

5.5 Сроки реализации проекта

Процесс разработки и сроки реализации ПО для поликлиник состоит из 6 этапов и включает в себя:

- сбор информации;
- проектирование и создание базы данных;
- разработка базы знаний;
- разработка клиентского приложения;
- тестирование программного обеспечения;
- оформление отчетов.

Этапы и сроки реализации проекта отображены в таблице 5.4.

Таблица 5.4 - Этапы и сроки реализации проекта

Перечень работ		Недели от начала работ									
		1	2	3	4	5	6	7	8	9	10
1 этап	Постановка задачи	■									
	Разработка схемы работы ПО	■	■								
	Подбор и изучение литературы		■	■							
	Подбор материала для заполнения информацией базы		■	■							
2 этап	Проектирование и создание базы данных			■	■						
	Заполнение базы данных				■	■					
3 этап	Создание базы знаний					■	■	■			
4 этап	Программирование графической оболочки						■	■	■		
5 этап	Тестирование ПО										■
	Отладка ПО										■
6 этап	Оформление										■
	Проверка и сдача отчета										■

5.6 Расчет стоимости работы по разработке

Расчет стоимости работы по разработке - это наиболее важная часть экономического анализа, так как на основе этого расчета определяются затраты рабочего времени на разработку проекта на каждом этапе, а также трудовые затраты.

Затраты на разработку данного проекта определяются по формуле

$$C = \text{ФОТ} + O_c + A + \text{Э} + C_{\text{пр}} + H \quad (5.1)$$

где ФОТ - фонд оплаты труда;

O_c - социальный налог;

A - амортизационные отчисления;

Э - затраты на электроэнергию;

$C_{\text{пр}}$ - прочие расходы;

H - накладные расходы.

5.6.1 Расчет затрат на оплату труда

Затраты на оплату труда персонала, задействованного в разработке проекта, рассчитываются по формуле:

$$\text{ФОТ} = \text{З}_{\text{осн}} + \text{З}_{\text{доп}} \quad (5.2)$$

где $\text{З}_{\text{осн}}$ - основная заработная плата;
 $\text{З}_{\text{доп}}$ - дополнительная заработная плата.

Труд программиста принят условно, на договорной основе в размере 150000 тенге.

Средний дневной заработок каждого работника рассчитывается по формуле:

$$D = \frac{\text{ЗП}_m}{D_p} \quad (5.3)$$

где ЗП_m – ежемесячный размер заработной платы;
 D_p – количество рабочих дней в месяце (это 22 дня – пятидневная рабочая неделя).

$$D = \frac{150000}{22} = 6818,2 \text{ тенге/день}$$

Заработная плата за один час вычисляется по формуле:

$$H = \frac{\text{ЗП}_m}{D_p * Ч_p} \quad (5.4)$$

где ЗП_m – ежемесячный размер заработной платы;
 D_p – количество рабочих дней в месяце.
 $Ч_p$ – Количество часов рабочего дня ($Ч_p = 8$)

$$H = \frac{150000}{22 * 8} = 852,3 \text{ тенге/час}$$

Длительность цикла в днях по каждому виду работ определяется по формуле:

$$t_n = \frac{T}{q_n * z * K} \quad (5.5)$$

где T – трудоемкость этапа, норма-час;
 q_n – количество исполнителей по этапу;
 z – продолжительность рабочего дня, $z = 8$ часов;
 K – коэффициент выполнения норм времени, $K = 1,1$.

Полученную величину t_n округляем в большую сторону до целых дней.

$$t_1 = \frac{24}{1 * 8 * 1,1} \gg 3 \text{ - программист, постановка задачи;}$$

Для каждого вида работ произведены аналогичные расчеты.

$$t_n = 3 + 3 + 3 + 4 + 6 + 3 + 4 + 12 + 2 + 2 + 1 + 1 = 44 \text{ дня}$$

Таким образом, для проведения всех работ необходимо 44 дня.

Сводные данные по расчету заработной платы персонала, задействованного в разработке проекта приведены в таблице 5.5.

Таблица 5.5 – Сводные данные по расчету основной заработной платы персонала задействованного в разработке проекта.

Наименование этапов	Исполнитель	Трудоемкость, норма-час	Длительность цикла, дни	Зарплата за час работы, тенге	Сумма заработной платы, тенге
Постановка задачи	Программист	24	3	852,3	20455,2
Разработка схемы работы ПО	Программист	24	3	852,3	20455,2
Подбор и изучение литературы	Программист	24	3	852,3	20455,2
Подбор материала для заполнения информации базы	Программист	32	4	852,3	27273,6
Проектирование и создание базы данных	Программист	48	6	852,3	40910,4
Заполнение базы данных	Программист	24	3	852,3	20455,2
Создание	Программист	32	4	852,3	27273,6

базы знаний					
Программирование графической оболочки	Программист	104	12	852,3	88639,2
Тестирование ПО	Программист	16	2	852,3	13636,8
Отладка ПО	Программист	16	2	852,3	13636,8
Оформление	Программист	8	1	852,3	6818,4
Проверка и сдача отчета	Программист	8	1	852,3	6818,4
Итого		360	44	10227,6	306828

Дополнительная заработная плата составляет 10% от основной заработной платы и вычисляется по формуле:

$$Z_{\text{доп}} = Z_{\text{осн}} * 0,1 \quad (5.6)$$

и составит

$$Z_{\text{доп}} = 306828 * 0,1 = 30682,8 \text{ тенге};$$

Таким образом, затраты на оплату труда согласно произведенным расчетам и в соответствии с формулой 5.2 составит

$$\text{ФОТ} = 306828 + 30682,8 = 337510,8 \text{ тенге}$$

5.6.2 Расчет затрат по социальному налогу

Социальный налог составляет 11% (ст. 358 п. 1 НК РК) от дохода работника, и рассчитывается по формуле

$$O_c = (\text{ФОТ} - \text{ПО}) * 11\% \quad (5.7)$$

где ПО – пенсионные отчисления, которые составляют 10% от ФОТ и социальным налогом не облагаются, вычисляются отчисления по формуле

$$\text{ПО} = \text{ФОТ} * 10\% \quad (5.8)$$

$$\text{ПО} = 337510,8 * 0,1 = 33751,08 \text{ тенге.}$$

Таким образом, в соответствии с произведенными расчетами и согласно формуле 5.7 размер отчислений на социальные нужды составит

$$O_c = (337510,8 - 33751,08) * 0,11 = 33413,6 \text{ тенге}$$

5.6.3 Расчет амортизационных отчислений

Амортизационные отчисления рассчитываются по формуле

$$A_i = \frac{H_A * C_{ПЕР} * N}{100 * n} \quad (5.9)$$

где H_A – норма амортизации;

$C_{ПЕР}$ – первоначальная стоимость оборудования;

N – количество дней на выполнение работ;

n – количество рабочих дней в году.

Норма амортизации на компьютерную технику составляет 40% от стоимости всего оборудования, на программное обеспечение - 15%.

Таким образом, амортизационный отчисления по используемому оборудованию, в соответствии с формулой 5.9 составят

- на ноутбук

$$A_1 = \frac{40 * 75000 * 44}{100 * 365} = 3616,44 \text{ тенге}$$

- на многофункциональное устройство

$$A_2 = \frac{40 * 30000 * 4}{100 * 365} = 131,5 \text{ тенге}$$

- на программное обеспечение

$$A_3 = \frac{15 * 85000 * 44}{100 * 365} = 1536,99 \text{ тенге}$$

$$A_i = 3616,44 + 131,5 + 1536,99 = 5284,93 \text{ тенге}$$

Сводные результаты расчета амортизационных отчислений представлены в таблице 5.6.

Таблица 5.6 – Сводные данные по расчету затрат на амортизацию

Наименование оборудования	Количество	Норма амортизации, %	Сумма амортизации, тенге
Ноутбук	1	40	3616,44
Многофункциональное устройство	1	40	131,5
Программное обеспечение	-	15	1536,99

Итого			5284,93
-------	--	--	---------

5.6.4 Расчет затрат на электроэнергию

Поскольку в процессе производства используется электрооборудование, необходимо рассчитать затраты на электроэнергию. Затраты на электроэнергию для производственных нужд включают в себя расходы электроэнергии на оборудование и дополнительные нужды. И рассчитываются по формуле

$$\mathcal{E} = \mathcal{Z}_{\text{эл.эн.об.}} + \mathcal{Z}_{\text{доп.}} \quad (5.10)$$

где $\mathcal{Z}_{\text{эл.эн.об.}}$ – затраты на электроэнергию для оборудования;
 $\mathcal{Z}_{\text{доп.}}$ – затраты на электроэнергию для дополнительных нужд.

Расходы электроэнергии для оборудования рассчитываются по формуле

$$\mathcal{Z}_{\text{эл.эн.об.}} = W * T * S * K_{\text{исп}} \quad (5.11)$$

где W – потребляемая мощность, Вт;

T – время работы, часы;

S – тариф (1кВт = 16,02 тенге);

$K_{\text{исп}}$ – коэффициент использования ($K_{\text{исп}} = 0,9$).

$$\mathcal{Z}_{\text{эл.эн.об. (ноутбук)}} = 0,7 * 360 * 16,02 * 0,9 * 1 = 3633,34 \text{ тенге}$$

$$\mathcal{Z}_{\text{эл.эн.об. (МФУ)}} = 0,9 * 32 * 16,02 * 0,9 = 415,24 \text{ тенге}$$

Общая сумма затрат на электроэнергию основного оборудования согласно формуле 4.11 составляет

$$\mathcal{Z}_{\text{эл.эн.об.}} = 3633,34 + 415,24 = 4048,58 \text{ тенге}$$

Затраты на дополнительные нужды берутся по показателю от затрат на оборудование в размере 5% и рассчитывается по формуле

$$\mathcal{Z}_{\text{доп.}} = \mathcal{Z}_{\text{эл.эн.об.}} * 5\% \quad (5.12)$$

и составляют

$$\mathcal{Z}_{\text{доп.}} = 4048,58 * 0,05 = 202,4 \text{ тенге}$$

Таким образом, суммарные затраты на электроэнергию, согласно формуле 5.11 составляют

$$\mathcal{E} = 4048,58 + 202,4 = 4250,98 \text{ тенге}$$

Сводные результаты расчета затрат на электроэнергию представлены в таблице 5.7

Таблица 5.7 – Сводные данные о затратах на электроэнергию

Наименование приборов	Количество	Потребляемая мощность, Вт	Число рабочих дней	Коэффициент использования	Время работы оборудования, часы	Сумма затрат, тенге
Ноутбук	2	0,7	44	0,9	360	3633,34
МФУ	1	0,9	4	0,9	32	415,24
Доп.затраты						202,4
Итого						4250,98

5.6.5 Расчет накладных и прочих расходов

Прочие расходы включают в себя:

1) расходы на интернет на 60 дней (2 месяца). Стоимость интернета за один месяц составляет 4500 тенге. Получаем:

$$4500 * 2 = 9000 \text{ тенге}$$

2) расходы на канцелярские товары:

- упаковка бумаги формата А4 – 1000 тенге;
- упаковка ручек – 300 тенге;
- 3 карандаша – 150 тенге;
- ластик – 50 тенге;
- линейка - 100 тенге;
- стикеры – 200 тенге.

Общая стоимость канцелярских товаров

$$1000 + 300 + 150 + 50 + 100 + 200 = 1800 \text{ тенге.}$$

3) аренда помещения на 60 дней (2 месяца).

Площадь помещения:

$$S = 3\text{м} * 4\text{м} = 12 \text{ м}^2.$$

Стоимость одного квадратного метра равна 1500 тенге. Получаем:

$$12 * 1500 * 2 = 36000 \text{ тенге.}$$

Прочие расходы составляют

$$C_{\text{ПР}} = 9000 + 1800 + 36000 = 46800 \text{ тенге.}$$

Накладные расходы составляют 50% от всех затрат и рассчитываются по формуле

$$(5.13) \quad H = (\text{ФОТ} + O_c + A + Э + C_{\text{ПР}}) * 50\%$$

$$H = (337510,8 + 33413,6 + 5284,93 + 4250,98 + 46800) * 0,5 =$$

$$= 213630,2 \text{ тенге}$$

5.6.6 Расчет стоимости по всем статьям затрат и определение структуры затрат

В соответствии с формулой 4.1 суммарные затраты по разработке моего проекта составляют

$$C = 337510,8 + 33413,6 + 5284,93 + 4250,98 + 46800 + 213630,2 =$$

$$= 640890,6 \text{ тенге}$$

Смета затрат по разработке программного обеспечения для медицинских учреждений, а также структура расходов представлены в таблице 5.8 и на рисунке 5.1

Таблица 5.8 – Суммарные данные по стоимости разработки проекта

Наименование статьи затрат	Сумма, тенге
Фонд оплаты труда	337510,8
Социальный налог	33413,6
Амортизация	5284,93
Затраты на электроэнергию	4250,98
Прочие расходы	46800
Накладные расходы	213630,2
Итого	640890,6

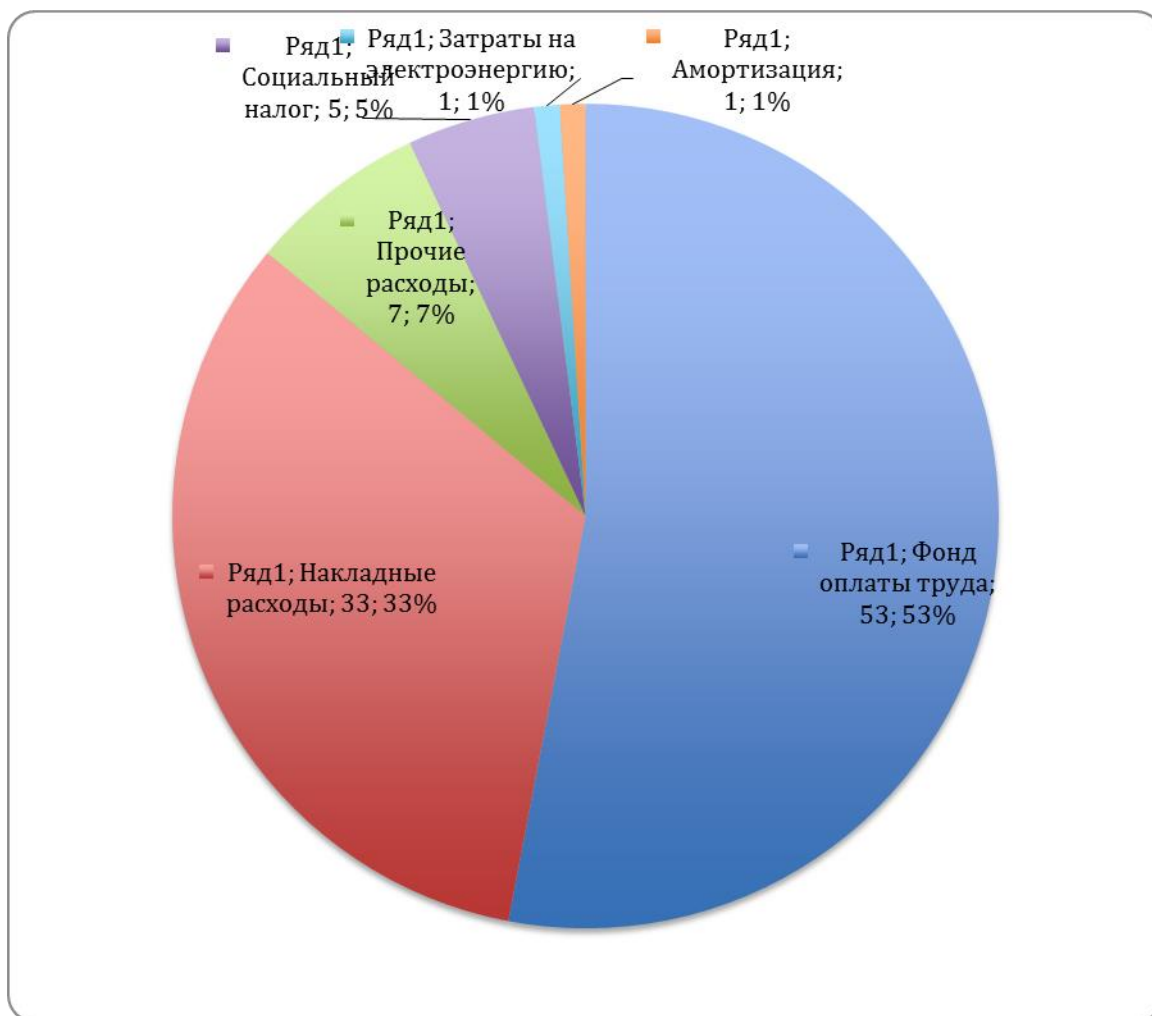


Рисунок 5.1 – Структура затрат по разработке сайта

5.7 Цена интеллектуального труда

Цена реализации проекта складывается из стоимости и чистого дохода

$$Ц = С + П \quad (5.14)$$

где С – стоимость продукта;
 П – чистый доход.

При определении первоначальной цены следует задаться желаемым уровнем рентабельности (здесь 20%) реализации программных продуктов

$$Ц_{П} = С * \left(1 + \frac{P}{100}\right) \quad (5.15)$$

где P – рентабельность.

$$Ц_{П} = 640890,6 * \left(1 + \frac{20}{100}\right) = 769068,72 \text{ тенге}$$

Цена реализации проекта рассчитывается по формуле

$$C_p = C_{\text{п}} + \text{НДС} \quad (5.16)$$

где НДС – налог на добавленную стоимость по ставке 12%

$$\text{НДС} = C_{\text{п}} * 12\% \quad (5.17)$$

$$\text{НДС} = 769068,72 * 0,12 = 92288,2 \text{ тенге}$$

В соответствии с формулой 5.16 цена реализации проекта составит

$$C_p = 769068,72 + 92288,2 = 861356,97 \text{ тенге}$$

В зависимости от спроса, окончательная цена на интеллектуальный труд в условиях рыночных отношений будет варьироваться и изменяться.

5.8 Вывод

Разработка программного обеспечения для медицинских учреждений является сложным и длительным процессом, требующим больших затрат умственного труда. Стоимость разработки включает в себя следующие категории расходов: вычеты из заработной платы на социальные нужды, амортизация, затраты на электроэнергию, а также прочие расходы.

Наибольшую долю в общей себестоимости разработки программного продукта составляют затраты на оплату труда в размере 337510,8 тенге, что составляет 53% от суммы разработки всего проекта.

Средняя цена услуг на реализацию такого проекта на рынке составила 2500000 [1] тенге. Эту цену мы получили, анализируя фонд оплаты труда различных компаний, занимающихся разработкой программного обеспечения, с учетом такого же количества работников и дней на реализацию проекта, как и у нас. Анализируя полученные расчеты, а точнее, полученную цену реализации проекта, равной 861356,97 тенге, можно сделать вывод о том, что данный проект является экономически выгодным.

6 Безопасность жизнедеятельности

6.1 Анализ условий труда

В настоящем дипломном проекте исследуется разработка программного обеспечения для медицинских учреждений с использованием технологии С#. Данная программа будет разрабатываться в офисе, в связи с чем должны быть учтены вопросы создания оптимальных условий.

В работе поставлены следующие задачи:

- а) рассмотреть факторы, определяющие безопасность здоровья программиста;
- б) подробно рассмотреть организационные факторы;
- в) произвести расчеты:
 - 1) уровня шума;
 - 2) молниезащиты;

В современных условиях работа в любой сфере деятельности сопровождается повышенной психофизической напряженностью, а также информационной и нервно-эмоциональной нагрузкой человека. Для того, чтобы ограничить либо устранить опасные факторы производства, решить вопросы организации труда, эргономичности помещений и рабочих мест, а также предотвратить чрезвычайные ситуации, необходимо провести ряд мероприятий, с помощью которых можно выявить факторы, способные нанести вред здоровью человека.

Повышенная температура внешней среды может также вызывать плохое самочувствие, хоть и в целом не нарушает здоровое состояние человека.

При работе на персональном компьютере, в помещении, соблюдаются следующие условия:

- а) Холодный период года:
 - 1) температура 19 – 24 °С;;
 - 2) относительная влажность 40 – 70 %;
 - 3) скорость движение воздуха относительная 0,2 м/с.
- б) Теплый период года:
 - 1) температура 25 – 28 °С;;
 - 2) относительная влажность 50 – 70;

3) скорость движение воздуха относительная 0,12 м/с.

Помещение представляет собой комнату с параметрами 8x4x3,2 м (Рисунок 1). Цвет стен – белые. В офисе применяется боковое естественное освещение. Также есть искусственное освещение в виде 4 светильников с люминесцентными лампами; электропитание: переменное напряжение 220-250 В, частотой 50 Гц, мощность светильника 2x65 Вт. Освещение помещения и оборудования должно быть мягким, без блеска. В помещениях, где находится компьютер, необходимо обеспечить следующие величины коэффициента отражения: для потолка: 60-70%, для стен: 40-50%, для пола: около 30%. Для других поверхностей и рабочей мебели: 30-40%.

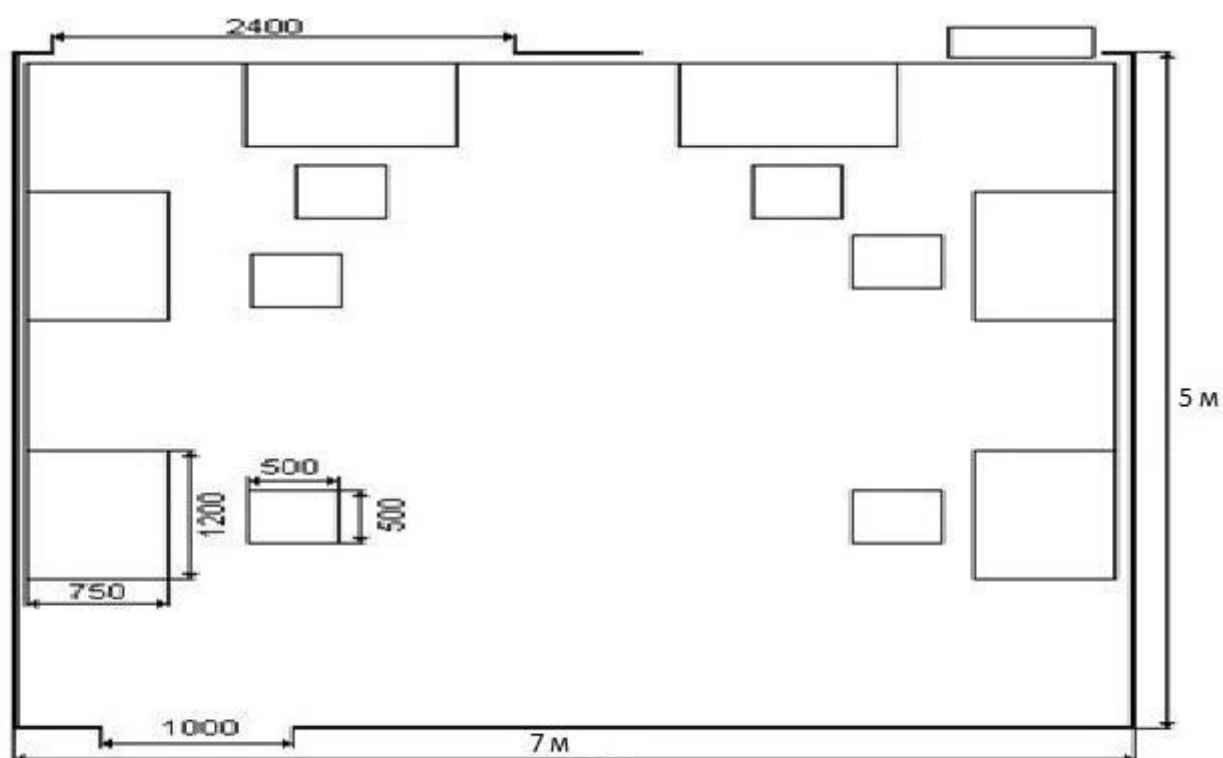


Рисунок 1 – План помещения

Рабочее место инженера-программиста оснащено следующим оборудованием:

- рабочий стол;
- стул;
- персональный компьютер в стандартной комплектации (системный блок, дисплей, клавиатура, манипулятор типа “мышь”);
- принтер.

Далее рассмотрим типичное офисное оборудование и степень его воздействия на работника.

1) Персональный компьютер.

Опасные и вредные производственные факторы, влияющие на человека, работающего с персональным компьютером, следующее:

- Повышенный уровень электромагнитного излучения;
- Повышенный уровень ионизирующего излучения;
- Повышенный уровень статического электричества;
- Увеличение интенсивности электростатического поля;
- Высокая или низкая ионизация воздуха;
- Повышенная яркость света;
- Прямой и отраженной блескосты;
- Высокое напряжение в электрической цепи, цепь, которая может произойти через тело человека;
- Статическая перегрузка опорно-двигательного аппарата и динамическую локальную перегрузку мышц рук;
- Перенапряжение зрительный анализатор;
- Психическое напряжение;
- Эмоциональные перегрузки;
- монотонность труда.

К вредным излучением компьютеров низкочастотных электромагнитных полей и ионизирующего излучения (рентгеновское) излучение мониторов на электронно-лучевых трубок (ЭЛТ).

Теперь несколько слов о радиации ЭЛТ-мониторов. В самом деле, этот вид излучения присутствует в любом устройстве, на электронно-лучевых трубок. Тем не менее, дистанцируясь на пятьдесят сантиметров от экрана монитора уровня излучения не превышает естественного фона. Кроме того, ЭЛТ-дисплеи почти полностью вытеснены жидкокристаллические дисплеи (LCD), в которых нет никакого излучения.

Есть более опасные риски при работе с персональными компьютерами. Все они не находятся в непосредственной опасности от этого вида техники и эргономических факторов, влияющих на здоровье человека.

Работа на ПК относится к визуальной напряженной работы. Значит, особенно работая с ПК ухудшается деятельность органов зрения. Именно этот вредный фактор присутствует в большинстве документов, регулирующих работу сотрудников с помощью компьютера.

Согласно СанПиН, есть три группы работы с видеодисплеями терминалами и ПК, а также три степени сложности и интенсивности труда. Первая группа работ (группа А) работа считывания информации с дисплея, вторая (группа В) - работа по вводу информации в третью (группа В) - интерактивную работу компьютера. Очевидно, что работа служащего в большинстве случаев относится к группе В, так как она требует интерактивной работы с различным программным обеспечением.

На зрительное утомление очень сильно зависит от уровня освещенности рабочего места. Это особенно заметно, когда необходимо одновременно

работать как с электронных и бумажных документов. Согласно СанПиН [3, пункт. 6.3.] Уровень освещенности на рабочем месте при работе на компьютере должна быть 300-500 лк. Монитор и источники света должны быть расположены таким образом, чтобы не создавать бликов на поверхности экрана.

Еще одна опасность таится в нашем офисе является статической позы при работе на компьютере. Статическое напряженная поза при длительной работе на компьютере может привести к воспалению мышц, сухожилий и связок спины и ног, заболевания позвоночника и суставов (в нижней части спины боли, тендинит и т.д.), а также постоянного напряжения рук - повредить запястья и сухожилия (так называемый синдром запястья или туннельный синдром). Эти болезни вызваны так называемыми травмами повторяющихся напряжений и постепенно накапливая болезни, вызванные длительным и многократным воздействием протекающей в болезни нервов, мышц и сухожилий. Воздействие этого неблагоприятного фактора снижается при правильной организации рабочего места - оптимально подобранной мебели, правильного расположения компонентов компьютера.

2) Принтер.

Если вы работаете с персональным компьютером в значительной степени регулируется законом, нельзя сказать об остальной части оборудования, используемого в офисе. Что касается принтера, то ясно: она не требует постоянного присутствия работника, фокус напряжения и т.д. Все работы с принтером сводится к нему и выключать, добавляя бумаги и удержание отпечатков. Даже замена картриджей не пользователь выполняет чаще и персонал.

Тем не менее, любой принтер является сложным прибор, поэтому при работе с ним вы должны выполнять все стандартные требования пожарной и электробезопасности. Кроме того, в любом принтере имеет движущихся частей, а также некоторые элементы могут нагреваться во время работы до высокой температуры (в матричных принтерах и струйной печатающей головки в лазерных принтерах - специальные нагревательные ролики).

Другим вредным фактором при работе с принтерами является шум. Самый высокий уровень шума при работе имеют матричных принтеров, но в паспортах этих устройств шума, как правило, не указано, но фактический уровень может быть измерен, за исключением оценки рабочего места. Это единственный тип принтера, уровень шума, который может быть сравнима с максимально допустимой на рабочем месте, оборудованном с ПК. Все остальные принтеры явно тише.

Шум ухудшает рабочую среду, обеспечивая вредное воздействие на организм человека. Работающие в долгосрочной перспективе воздействия шума испытали раздражительность, головные боли, головокружение, потеря памяти, повышенная утомляемость, потеря аппетита, боль в ушах, и так

далее. D. Такие нарушения в ряде органов и человеческого организма может вызвать негативные изменения в эмоциональном состоянии человека до стресса. Под влиянием шума снижается концентрация внимание нарушаются физиологические функции появляется усталость в связи с затратами на энергию и психоневрологических напряжением, ухудшается функция голоса. Все это снижает эффективность четырех-прав и его производительность, качество и безопасность. Долгое воздействия интенсивного шума [выше 80 дБ (А)] для человеческого уха приводит к его частичной или полной потери.

Уровень шума на рабочем месте не должен превышать 50 дБА. Для снижения уровня шума стены и потолок помещений, где установлены компьютеры, могут быть облицованы звукопоглощающими материалами. Уровень вибрации в помещениях может быть снижен путем установки оборудования на специальные виброизоляторы.

Расчетная часть

6.2 Расчет уровня шума

Одним из негативных факторов рабочей среды является высокий уровень шума, производимого печатающих устройств, воздуха кондиционирование оборудования, вентиляторов охлаждения в самих компьютерных системах.

Для того, чтобы рассмотреть вопрос о необходимости и целесообразности снижения шума необходимо знать уровни шума на рабочем месте.

Шум, возникающий от нескольких некогерентных источников работающих одновременно, рассчитывается на основе принципа энергии суммирования отдельных источников излучения:

$$L_{\Sigma} = 10 \lg \sum_{i=1}^{i=n} 10^{0,1L_i},$$

где L_i – уровень звукового давления i -го источника шума;
 n – количество источников шума.

Полученные результаты расчета сравниваются с допустимым уровнем значения шума для данного рабочего места. Если результаты расчета выше допустимого значения уровня шума, что необходимы специальные меры для снижения уровня шума. К ним относятся: облицовка стены и потолок помещения звукопоглощающих материалов, уменьшение шума в источнике, собственно макет оборудования и рациональной организации рабочего места оператора.

Уровни звукового давления источников шума, действующих на оператора на его рабочем месте представлены в табл. 1.2.

Таблица 1.2 Уровни звукового давления различных источников.

Источник шума	Уровень шума, дБ
Жесткий диск	40
Вентилятор	45
Монитор	17
Клавиатура	10
Принтер	45
Сканер	42

Обычно рабочее место оператора оснащено следующим оборудованием: винчестер в системном блоке, вентилятор(ы) систем охлаждения ПК, монитор, клавиатура, принтер и сканер.

Подставив значения уровня звукового давления для каждого вида оборудования в формулу, получим:

$$L_{\Sigma} = 10 \cdot \lg (10^4 + 10^{4.5} + 10^{1.7} + 10^1 + 10^{4.5} + 10^{4.2}) = 49,5 \text{ дБ}$$

Полученное значение не превышает допустимый уровень шума для рабочего места оператора, равный 65 дБ (ГОСТ 12.1.003-83). И если учесть, что вряд ли такие периферийные устройства как сканер и принтер будут использоваться одновременно, то эта цифра будет еще ниже. Кроме того при работе принтера непосредственное присутствие оператора необязательно, т.к. принтер снабжен механизмом автоподачи листов.

6.3 Расчет молниезащиты офисного помещения одиночным стержневым молниеотводом.

Молниеотвод необходим для защиты общественных, жилых а также производственных зданий от удара молнии. Принимая на себя электрический разряд молнии, данное устройство отводит ток молнии в землю.

Молния состоит из опорной части (поддержки), молниеприемника, коллектора и заземления. Самый общий стержень и кабель коллектора и заземления. Самый распространенный стержень молнии и кабель. Каждый громоотвод имеет определенную область защиты пространства, внутри которого защищенные строительные конструкции от прямых ударов молнии.

При прохождении тока молнии через молниеотвода на основе потенциальной опасных для человека может происходить в непосредственной близости от земли.

Защита от молнии характеризуется зоной, которая определяется как часть пространства охраняемого удара молнии с определенной степенью надежности. Согласно Приложению 3. Зоны защиты молниеотводов Инструкции по устройству молниезащиты зданий и сооружений, исходя из показателей безопасности зоны защиты делятся на 2 вида – тип А и тип Б. Тип зоны защиты выбирают исходя из возможного показателя поражений молнией зданий и сооружений в год (N). Когда показатель равен $N > 2$, то принимают зону защиты типа А (степень надежности защиты в этом случае составляет не менее 99.5%). При $0,01 < N \leq 2$ принимают зону защиты типа Б

(степень надежности этой защиты - 95% и выше).

Существует три категории зданий и сооружений, в данном дипломном проекте рассматривается здание, относящееся к III категории, т.к. относится к общественным зданиям III, IV, V степени огнестойкости, согласно Правилам устройства электроустановок (ПУЭ-76).

Размеры объекта:

длина $A=15$;

ширина $B=7$;

высота $H=15$;

Молниеотвод устанавливается в центре крыши здания. Схема зоны защиты здания одиночным стержневым молниеотводом представлена на рисунке 1.

Среднегодовая продолжительность гроз в часах определяется по региональным картам продолжительности гроз. Подсчет ожидаемого количества N поражений молнией в год производится по формуле:

$$N = (A + 6h_x)(B + 6h_x)n * 10^{-6} \quad (6.1)$$

, где h - наибольшая высота здания или сооружения, м;

B, A - соответственно ширина и длина здания или сооружения, м;

n - среднегодовое число ударов молнии в 1 км земной поверхности.

Таблица 6.3 Удельные плотности ударов молнии для различных среднегодовых показателей гроз

Среднегодовая продолжительность гроз, ч	Удельная плотность ударов молнии в землю n , 1/(км ² •год)
10-20	1
20-40	2
40-60	4
60-80	5,5
80-100	7
100 и более	8,5

Так как среднегодовая продолжительность гроз в области месторасположения здания составляет 20-40 часов, то удельная плотность ударов молнии в землю будет равна 2, согласно таблице 6.3.

По схеме защиты (см. рисунок 2) горизонтальное сечение зоны защиты R_x представляет половину диагонали защищаемого здания:

$$R_x = \sqrt{(A/2)^2 + (B/2)^2} \quad (6.2)$$

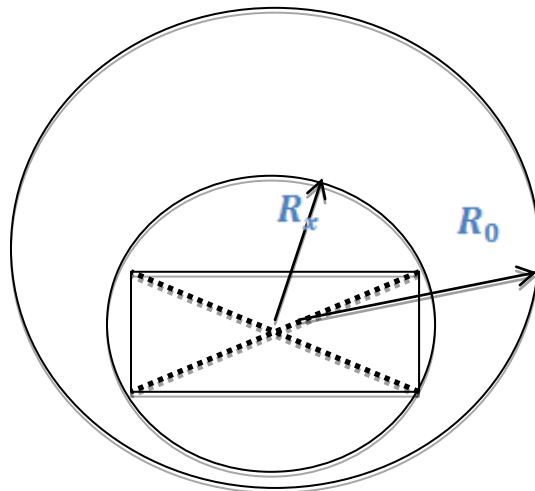
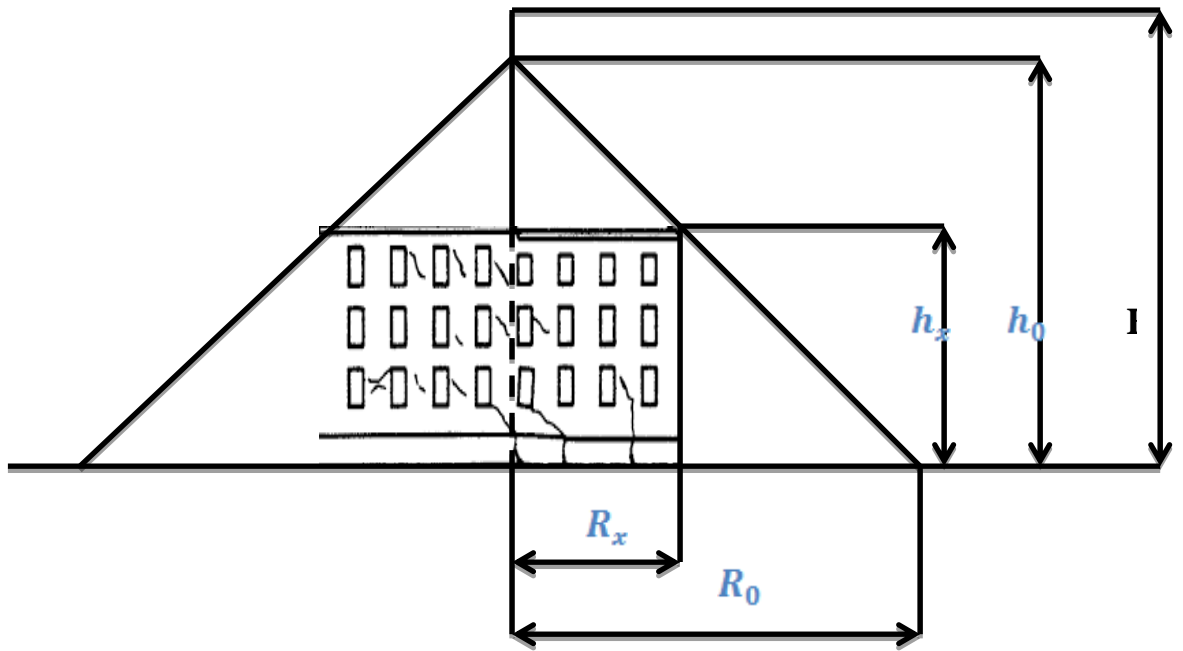


Рисунок 2 – Схема зоны защиты здания одиночным стержневым молниеотводом

Подставив значения в формулу (6.2), получим

$$R_x = \sqrt{(15/2)^2 + (7/2)^2} = 8,28 \text{ м. Принимаем } 8 \text{ м.}$$

Тогда, исходя из формулы (1.1)

$$N = (15 + 6 * 15)(7 + 6 * 15)2 * 10^{-6} = 0,02.$$

Так как $0,01 < N \leq 2$, принимается тип зоны защиты молниеотвода – Б.

Минимально расстояние между молниеотводом и объектом принимаем равным нулю, т.к. исходя из данных раздела 4 ПУЭ, допускается установка на крыше зданий и сооружений II и III категории по молниезащите.

Основные параметры типы зоны Б рассчитываются по формулам:

$$h_0 = 0,92h;$$

$$R_0 = 1,5h;$$

$$R_x = 1,5(h - h_x/0,92);$$

$$h = (R_x + 1,63 h_x)/1,5.$$

Согласно имеющимся данным, последовательно вычисляем искомые параметры:

$$h = (8 + 1,63 * 15)/1,5 = 21,63 \approx 22 \text{ м};$$

$$h_0 = 0,92 * 22 = 20,24 \text{ м};$$

$$R_0 = 1,5 * 22 = 33 \text{ м}.$$

Как видно по проделанным вычислениям, в центре крыши здания необходимо установить стержневой молниеотвод на опоре высотой 15 м, при этом радиус горизонтального сечения зоны защиты на высоте здания составит 8 м, а на уровне земли зона защиты образует круг радиусом 33 м, общая высота молниеотвода - 22 м

Выводы

Выполнен анализ условий труда в производственном помещении, признанный допустимым, и проверенный расчетами уровня шума и вибрации, а также молниезащиты с помощью одиночного стержневого молниеотвода для офисного помещения. По результирующим данным расчета уровня шума и вибрации было проверено, что оборудование, с данным уровнем звукового давления, используемое при разработке, может гарантировать приемлемые условия труда в офисе.

Заключение

В ходе выполнения дипломной работы было разработано программное обеспечение для типовой поликлиники. Программа реализована с использованием языка программирования С#, технологии ADO.NET.

Программное обеспечение является примером реального корпоративного приложения, функционирующих на базе сетевых технологий.

Используя архитектуру веб-клиент, разработанный подсистема предоставляет пользователям различных компьютеров доступ к серверу для необходимой информации. Сервер предоставляет возможность клиенту работать с базой данных. Он действует как посредник между клиентом и базой данных. Он получает запросы от клиента, обрабатывает их и посылает к основанию данных. Также он направляет запрос обратно клиенту.

Клиент имеет возможность работать с информацией, хранящейся в базе данных. Эта информация представлена в виде таблиц базы данных. Клиент может просматривать, редактировать, поиск и т.д. необходимую ему информацию.

Использование базы данных в качестве хранилища информации оптимально и эффективно хранить информацию, ее структуру.

Кроме того, в результате этого дипломного проекта мы получили дополнительные навыки в работе с С#, UML, были получены знания о различных методах построения функциональных и информационных моделей.

Во время тестирования разработанного программного обеспечения было установлено, он работает правильно и соответствует изложенным функциональным требованиям. В целом, можно предположить, что цель дипломной работы достигнута.

Список литературы:

- 1) <http://web.it-sfera.ru>
- 2) Сайт <http://www.fireman.kz/content/359.html>
- 3) <http://electricalschool.info/books/504-puje-7-pravila-ustrojstva.html>
- 4) Сайт о инженерных коммуникациях <http://www.ing-seti.ru/?p=167>
- 5) Медицинские информационные системы. Теория и практика / Под редакцией Г.И. Назаренко и Г.С. Осипова. Москва ФИЗМАТЛИТ 2005.
- 6) [Айламазян и Гулиев, 2000] *Айламазян А.К., Гулиев Я.И.* Разработка информационных систем лечебно-профилактических учреждений: проблемы и решения // Тез. докл. Международного форума «Информатизация процессов охраны здоровья населения — 2000». — М., 2000.
- 7) [Айламазян и Гулиев, 2001] *Айламазян А.К., Гулиев Я.И.* Данные, документы и архитектура медицинских информационных систем // Тез. докл. Международного форума «Информатизация процессов охраны здоровья населения — 2001». — М., 2001. — С. 141–142.
- 8) [Айламазян и др., 1999]
- 9) *Айламазян А.К., Гулиев Я.И., Комаров С.И., Малых В.Л., Морозов В.Ю.* Информационные системы в медицине: проблемы и решения // Программные системы: Теоретические основы и приложения / Под ред. А. К. Айламазяна. — М.: Наука. Физматлит, 1999. — С. 162–168.
- 10) http://sitemonitor.ru/doc/UML_HTML/gl_17.htm
- 11) <http://www.intuit.ru/studies/courses/1007/229/lecture/3178?page=3>
- 12) Сатимова Е.Г. Проектирование баз данных. Методические указания к выполнению лабораторных работ (для студентов всех специальностей). – Алматы: АИЭС, 2009
- 13) Питер Роб, Карлос Коронел Системы баз данных: проектирование, реализация и управление, 5-е издание, – ВHV Санкт-Петербург, 2004 .
- 14) Плю Р., Стефенс Р., Райан К. Освой самостоятельно SQL за 24 часа. – М.: Издательский дом «Вильямс», 2000.
- 15) <http://www.sql.ru/forum/829671/kopirovanie-v-bufer-koda-transact-sql>
- 16) <http://www.piter.com/collection/yazyk-programmirovaniya-ss-s/product/yazyk-programmirovaniya-c-klassika-computers-science-4-e-izd>

Приложение А

Листинг программы

```
Program.cs
using System;
using System.Collections.Generic;
using System.Linq;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace DanaDiplomaProject
{
    static class Program
    {
        /// <summary>
        /// Главная точка входа для приложения.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace DanaDiplomaProject
{
    public partial class Form6 : Form
```

```

{
    public Form6()
    {
        InitializeComponent();
    }

    private void label1_Click(object sender, EventArgs e)
    {

    }

    private void label7_Click(object sender, EventArgs e)
    {

    }

    private void Form6_Load(object sender, EventArgs e)
    {

    }
}

```

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

```

```

namespace DanaDiplomaProject
{
    public partial class Form3 : Form
    {
        public Form3()
        {
            InitializeComponent();
        }

        private void label1_Click(object sender, EventArgs e)
        {

```

```

    }

    private void button1_Click(object sender, EventArgs e)
    {
        Form6 x = new Form6();
        x.ShowDialog();
    }

    private void textBox7_TextChanged(object sender, EventArgs e)
    {

    }

    private void Form3_Load(object sender, EventArgs e)
    {

    }
}

```

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

```

```

namespace DanaDiplomaProject
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            if (this.textBox1.Text.ToString().Contains('1'))
            {
                MessageBox.Show("Неверное значение!");
            }
        }
    }
}

```

```

        else
        {
            this.Hide();
            Form3 x = new Form3();
            x.ShowDialog();
        }
    }

    private void label1_Click(object sender, EventArgs e)
    {

    }

    private void Form2_Load(object sender, EventArgs e)
    {

    }

    private void button2_Click(object sender, EventArgs e)
    {
        Form6 x = new Form6();
        x.ShowDialog();
    }

    private void textBox1_TextChanged(object sender, EventArgs e)
    {

    }

    private void textBox2_TextChanged(object sender, EventArgs e)
    {

    }

    }
}

```

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

```

```

using System.Windows.Forms;

namespace DanaDiplomaProject
{
    public partial class Form1 : Form
    {
        string passDoctor = "123";
        string passReg = "1234";
        string passLab = "12345";

        public Form1()
        {
            InitializeComponent();
        }

        private void label2_Click(object sender, EventArgs e)
        {
        }

        private void button1_Click(object sender, EventArgs e)
        {
            switch (comboBox1.SelectedIndex)
            {
                case 1:
                    if (passwordField.Text.ToString() == this.passDoctor)
                    {
                        this.Hide();
                        Form2 x = new Form2();
                        x.ShowDialog();
                    }
                    else {
                        MessageBox.Show("Неверный пароль!");
                    }

                    break;

                case 0:
                    if (passwordField.Text.ToString() == this.passReg)
                    {
                        Form4 x = new Form4();
                        x.ShowDialog();
                    }
            }
        }
    }
}

```

```

    }
    else
    {
        MessageBox.Show("Неверный пароль!");
    }
    break;
case 2:
    if (passwordField.Text.ToString() == this.passLab)
    {
        Form5 x = new Form5();
        x.ShowDialog();
    }
    else
    {
        MessageBox.Show("Неверный пароль!");
    }
    break;
}
}

private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
}

private void passwordField_TextChanged(object sender, EventArgs e)
{
    // Set to no text.

    // The password character is an asterisk.
    passwordField.PasswordChar = '*';
    // The control will allow no more than 14 characters.
    passwordField.MaxLength = 14;
}

private void contextMenuStrip1_Opening(object sender, CancelEventArgs e)
{
}

```

```
    }  
    private void Form1_Load(object sender, EventArgs e)  
    {  
    }  
}
```