

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество  
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

кафедра Компьютерных технологий

«Допущен к защите»  
Заведующий кафедрой \_\_\_\_\_  
Курманбаев З.К.  
(Ф.И.О., ученая степень, звание)

« \_\_\_\_\_ » \_\_\_\_\_ 20\_\_ г.  
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Разработка программного обеспечения для исследовательских учреждений с использованием технологии С# (часть 1)

Специальность 5В070400 - Вычислительная техника и программное обеспечение

Выполнил (а) Байбек Дана ВТ-12-3  
(Фамилия и инициалы) группа

Научный руководитель К.ф.-и.н. доцент Шайхин Б.М.  
(Фамилия и инициалы, ученая степень, звание)

Консультанты:

по экономической части:

Бексатова А.У., к.т.н., доцент  
(Фамилия и инициалы, ученая степень, звание)  
А.У. « 30 » 05 2016 г.  
(подпись)

по безопасности жизнедеятельности:

Бригадиров Н.Б., в.х.н., доцент  
(Фамилия и инициалы, ученая степень, звание)  
Н.Б. « 02 » 06 2016 г.  
(подпись)

по применению вычислительной техники:

Шайхин Б.М., к.ф.-и.н. доц.  
(Фамилия и инициалы, ученая степень, звание)  
Б.М. « 28 » 05 2016 г.  
(подпись)

Нормоконтролер: Шайхин Б.М. к.ф.-и.н. доц.  
(Фамилия и инициалы, ученая степень, звание)  
Б.М. « 02 » 06 2016 г.  
(подпись)

Рецензент: Серван В.К. к.т.н.  
(Фамилия и инициалы, ученая степень, звание)  
В.К. « 06 » 06 2016 г.  
(подпись)

Алматы 2016 г.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество  
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Факультет Аэрокосмических и информационных Технологий  
Специальность Водосливательная техника и механические системы  
Кафедра Компьютерных технологий

ЗАДАНИЕ

на выполнение дипломного проекта

Студент Баубек Дана  
(фамилия, имя, отчество)

Тема проекта Разработка программного обеспечения для медицинских учреждений с использованием технологии С# (часть 1)

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

Исходные материалы С#, база знаний интеллектуальной системы. Пособие

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

1. Определение предметной области
2. Анализ технологии
3. База знаний
4. Структура БД
5. Разработка интерфейса
6. Расчет технико-экономического обоснования
7. Описание безопасности и надежности

Перечень графического материала (с точным указанием обязательных чертежей)

Рисунки технологий Visual Studio, MS SQL Server  
 Рисунки диаграмм UML  
 Рисунки интерфейса ПО  
 Таблица экономического обоснования  
 Рисунки помещений и размещения конди-  
 ционера и подводящих сетей

Рекомендуемая основная литература

Дин. Рихтер, CLR via C#. Программирование на  
 платформе Microsoft .NET Framework 4.5 на  
 языке C#. 4-е изд.  
 Сайт: <http://habrahabr.ru>  
 Сайт: <http://helpcenter.onlyoffice.com/ru/guides/knowledgebase.aspx>  
 Табишова - База знаний интеллектуальных систем.  
 Москва, 2003. - С. 234.

Консультанты по проекту с указанием относящихся к ним разделов

Раздел	Консультант	Сроки	Подпись
Экскл. часть БЖД	Бекмурза А.Ф.	20.05.-30.05.16	[Подпись]
Раздел 1	Природово И.Г.	18.03.-02.06.16	[Подпись]
Раздел 2	Шайхин Б.М.	22.02.-18.03.16	[Подпись]
Раздел 3	Шайхин Б.М.	28.03.-18.04.16	[Подпись]
Модуль	Шайхин Б.М.	19.04.-26.05.16	[Подпись]
		02.06.16	[Подпись]



## **Аңдатпа**

Дипломдық жұмыста, С# технологиясы көмегімен медициналық мекемелерге арналған, біріншілік медициналық санитарлық көмек көрсететін мамандардың жұмысын оңтайландыратын жүйе жобаланған. Жұмыс Visual Studio жобалау ортада, С# бағдарламалау технологиясы негізінде және MS SQL Server мәліметтер базасы мен ASP.NET сияқты қосымша технологияларды қолдана отырып орындалған.

Дипломдық жұмыстың өзектілігі - бұл жүйе замануы және бүгінгі таңда осы бағдарламалық қамтамасыздықтың баламасы жоқ. Ол науқастарды қабылдауды оңтайландырады, науқастар туралы маңызды ақпараттарды мәліметтер базасында сақтауға мүмкіншілік береді, ақпараттарды өңдейді және науқастың алғашқы қаралуында оған қажетті тексерулер тізімін өңдеп береді. Жұмыста өмір сүру қауіпсіздігінің шарттары және техникалық-экономикалық дәлелдемесі ескерілген.

## **Аннотация**

В дипломной работе была спроектирована система для медицинских учреждений с помощью технологии С#, которая оптимизирует работу специалистов, оказывающих первичную медико-санитарную помощь. Работа выполнена на основе технологии программирования С# в среде разработки Visual Studio с использованием сопутствующих технологий, таких как база данных MS SQL Server и ASP.NET.

Актуальность дипломной работы заключается в том, что данная система является современной и на сегодняшний день нет аналогов данного программного обеспечения, которое улучшает прием пациентов, позволяет хранить важную информацию о пациенте в базе данных, производит обработку информации и выдает перечень необходимых обследований для пациента при первичном обращении. В работе были учтены меры безопасности жизнедеятельности и технико-экономическое обоснование.

## **Annotation**

In this diploma project has been designed the system for the medical institutions using C# technology, which optimizes the work of specialists, providing primary health care. The work is done on the basis of the C# programming techniques in Visual Studio development environment with related technologies such as database MS SQL Server and ASP.NET.

The relevance of the diploma project is that the system is modernized, and today there are no analogues of this software, which improves the patients receiving, allows keeping important information about a patient in the database, processes the information and produces a list of required surveys for the patient in the primary treatment. The work has been taken into account health and safety measures and the feasibility study.

## Содержание

Введение .....	8-9
1 Принципы разработки .....	10-14
1.1 Анализ текущей ситуации .....	10-11
1.2 МИС «Medwork» .....	11-12
1.3 Технологии разработки .....	13-14
1.3.1 Microsoft Visual Studio .....	13
1.3.2 С# .....	14
1.3.3 Microsoft SQL Server .....	14
1.3.4 ASP.NET .....	14
2 База знаний .....	15-23
2.1 Парсинг программы .....	18-23
3 Структура разработки .....	24-35
3.1 Выбор и обоснование алгоритма программы .....	24
3.2 Моделирование ИС .....	24-31
3.2.1 Моделирование БД .....	26
3.2.2 Диаграммное представление .....	26-27
3.2.3 Основные понятия модели ER (Сущность-Связи) .....	27-28
3.2.4 Диаграммы .....	28-31
3.3 Реализация программы .....	32-35
4 Техничко-экономическая часть .....	36-47
4.1 Описание работы и обоснование необходимости .....	36
4.2 Трудовые ресурсы, используемые в работе .....	36-37
4.3 Оборудование, используемое в работе .....	37
4.4 Программное обеспечение, используемое в работе .....	37-38
4.5 Сроки реализации проекта .....	38
4.6 Расчет стоимости работы по разработке .....	39
4.6.1 Расчет затрат на оплату труда .....	39-42
4.6.2 Расчет затрат по социальному налогу .....	42

4.6.3 Расчет амортизационных отчислений .....	42-43
4.6.4 Расчет затрат на электроэнергию .....	43-44
4.6.5 Расчет накладных и прочих расходов .....	45
4.6.6 Расчет стоимости по всем статьям затрат и определение структуры затрат .....	45-46
4.7 Цена интеллектуального труда .....	46-47
4.8 Вывод по технико-экономическому обоснованию .....	47
5 Безопасность жизнедеятельности .....	48-58
5.1 Анализ условий труда .....	48-53
5.2 Расчет системы кондиционирования .....	53-55
5.3 Пожарная безопасность .....	55-58
5.4 Вывод по части безопасности жизнедеятельности .....	58
Заключение .....	59
Список используемой литературы .....	60
Приложение А .....	61-67

## Введение

Ни для кого не секрет какие огромные очереди приходится отстаивать людям в поликлиниках нашей страны. Для начала, давайте рассмотрим причины возникновения очередей. По данным мониторинга, проведенного министерством здравоохранения Республики Казахстан, 40–45% обращений в поликлиники ведут к тому, что граждане хотят получить справку или разъяснения по рецептам. Вторая категория лиц – «самообращающиеся», которые сдают анализы или проходят медосмотр. Они традиционно не записываются и создают "живую" очередь. И третья категория – это лица, которые состоят на учете и должны по определенному, установленному законодательством режиму, приходить на прием к врачу [1]. Вдобавок, существуют пробелы с недостатком кадров. Конечно, в стране реализуют программы по снижению очередей с помощью развития системы электронных очередей, но этого недостаточно.

Сделав опрос среди врачей поликлиник города Алматы, я пришла к выводу, что эффективным способом будет, если открыть кабинет первичного приема, оснащенного современной системой информационной технологии. В кабинете могут работать как врачи, например, начинающие (интерны, резиденты), так и специалисты сестринского дела с высшим образованием. Они смогут производить первичный осмотр пациентов по основным жизненным параметрам, вводить эти информации в базу данных; программное обеспечение могло бы переработать эти данные и выдало бы предварительный перечень необходимых обследований. Сотрудник кабинета первичного приема, на основании данного перечня, сможет направить их на сдачу анализов, после чего записать к соответствующим специалистам.

На сегодняшний день главными и важными составляющими в работе медицинских учреждений являются учет, контроль и управление. Автоматизирование медицинских учреждений сможет создать единую и прочную систему. Данная система представляет собой компьютерную программу, которая оптимизирует работу медицинских учреждений, оказывающих амбулаторный прием. Одним словом, облегчит прием пациентов, сократив огромные очереди, уменьшит время, затрачиваемое на осмотр пациента, что ведет к повышению уровня всей работы учреждения в целом.

Каким должно быть программное обеспечение? Конечно же, простым и удобным в использовании. Ввиду этого, в данной дипломной работе мое программное обеспечение спроектировано на языке C#, с использованием баз данных MS SQL Server. C# - гибкий, изящный и предельно практичный язык программирования. В ходе работы было разработано программное обеспечение, которое будет понятным для всех, ведь у людей, далеких от IT технологий, не будет времени сидеть и разбираться с программой.



Требования к программному продукту:

- Хранение информации в компактном электронном виде
- Ведение электронной истории болезней
- Выдача необходимого количества анализов
- Возможность лаборатории принимать заявки на анализы и передавать результаты в общий электронный банк данных
- Выдача справки - заключения с листом назначения

Дипломная работа состоит из 5 частей.

Первый раздел состоит из теоретического введения и обзора на использованную технологию.

Второй раздел состоит из базы знаний, а именно из использованных правил, из алгоритмов поиска.

Третий раздел состоит из схемы базы данных и реализации программного обеспечения.

Четвертый раздел состоит из расчетов технико-экономического обоснования реализации программного обеспечения, где в итоге вычислила себестоимость разработки.

Пятый раздел состоит из анализов вредных производственных факторов, вызванных реализацией программы, влияющих на сотрудников, расчеты системы кондиционирования и пожаробезопасность.

## **1 Принципы разработки**

### **1.1 Анализ текущей ситуации**

На сегодняшний день в целях увеличения эффективности Правительства РК разработаны и внедрены такие проекты, как электронный документооборот, электронные государственные закупки, порталы государственных органов, на подходе проект е-здравоохранение, включающий все лечебные учреждения и всю систему управления медициной Казахстана.

Но все же, имеет место быть проблема бюрократизации, что в свою очередь оказывает влияние на качество обслуживания пациентов. С каждым годом увеличивается количество очередей во врачебные кабинеты, у пациентов нет широкой альтернативы при выборе врача, к тому же не могут заранее планировать время для приема у врача. Все эти проблемы разрешимы путём внедрения ИКТ, перехода к «электронному здравоохранению», что позволяет увеличить доступность медицинских услуг для народа, сократить время на заполнение бумаг, а взамен использовать это время непосредственно для обслуживания пациентов, обеспечить оперативность и точность диагностики и лечения пациентов.

Система данной дипломной работы призвана оптимизировать работу медицинских учреждений, создать общее информационное пространство, повысить эффективность работы персонала, позволит уменьшить бумажный документооборот, увеличить скорость доступа к информации, улучшить эффективность работы медицинских учреждений в целом.

### **1.2 МИС «Medwork»**

Данная медицинская система производит управление поликлиникой, больницей, аптекой, медицинским центром и дает возможность организовать работу в виде единого механизма всех работников по медицинскому учреждению. Система позволяет сотрудникам организации открывать электронные истории болезней на регистратуре, проводить учет медицинских книжек и записывать пациентов на прием. Медицинская программа мониторинга дает возможность присматривать кассирам за своим рабочим местом, записывать на услугу, а также принимать оплату от граждан. Автоматизированная медицинская работа дает возможность специалистам разного рода профилей, видеть в открытой программе тех, кто записан к ним на прием. Медицинское приложение может хранить данные в течение многих лет, и это гораздо проще для поиска информации, чем на бумаге. Медицинское программное обеспечение позволяет автоматически заполнять медицинские записи, книги журналы и формы обязательных медицинских отчетов. Используя медицинскую систему, с рабочего места можно управлять больницами и поликлиниками. В связи с этим, вести медицинские записи

становится невероятно легко. По желанию, система медицинской документации может быть дополнена аналитическим учетом для руководителей здравоохранения. С помощью медицинской программы можно анализировать данные в контексте служб. Программа статистики в области здравоохранения может быть дополнена учетом хранения лекарственных средств, продуктов питания для столовой, руководством аптеки. Государственная медицинская программа полностью поддерживает автоматизацию бухгалтерского учета в соответствии с международными стандартами в области здравоохранения. Управление в области медицины, программы медицинской документации, медицинской автоматизации - все это не только очень удобно, это также является показателем уровня медицинских учреждений, формирования отношения пациентов и мнение других сотрудничающих организаций.

### **1.3 MedElement**

MedElement является "облачным" сервисом и справочной системой для врачей и медицинских организаций, а также просто для тех, кто заботится о своем здоровье.

Целью программы MedElement является как повышение качества медицинских услуг, так и улучшение здоровья населения.

Основные преимущества:

- Облако хранения - доступ к информации и веб-службам из любой точки мира
  - Легкая связь между клиникой и пациентом посредством удобных веб-сервисов
  - Обширная БД медицинской информации, собранной на одной платформе
- Врачам:
- Электронный кабинет врача представляет собой интернет-сервис для удобного учета пациентов и постановки диагноза
  - Медицинские справочники электронного типа
  - Толкование лабораторных показателей
  - Бесплатные мобильные приложения (отзывы о врачах и клиниках, справочник заболеваний, база клинических протоколов)

Клиникам:

MedElement дает эффективные решения по управлению медицинскими учреждениями:

- автоматизация клиники через интернет быстрым и недорогим способом (облачная медицинская информационная система, МИС)
- автоматизация производственных запасов бухгалтерского учета и финансов в клинике (амбулаторная и стационарная)

- автоматизация общественного питания (больницы, обслуживающий персонал)
- каталог медицинских организаций (Казахстан) - Быстро создать шаблон сайта-визитки, отзывы пациентов
- поиск информации о заболевании. Повышение уровня информированности о медицинских услугах и государственных стандартов диагностики и лечения заболеваний
- занимательная информация для пациентов (как измерить рост, вес, артериальное давление и других параметров)
- бесплатные мобильные приложения (отзывы врачей и больниц, стандартных условиях, база клинических протоколов)

Особенности:

- надежная информация из достоверных источников (научная литература, актуальные публикации, государственные стандарты, статистические данные)
- непрерывное обновление справочных БД
- корреляция различных разделов справочной системы

## 1.4 Технологии разработки

**1.4.1 Microsoft Visual Studio** - один из продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Эти продукты помогают разрабатывать консольные приложения, приложения с графическим интерфейсом с поддержкой технологии Windows Forms, веб-сайты, веб-приложения для всех платформ, поддерживаемых Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Silverlight.

Visual Studio содержит редактор исходного кода с поддержкой технологии IntelliSense и может делать простейший рефакторинг кода. В возможности встроенного отладчика входят: работа отладчика уровня исходного кода, и работа отладчика машинного уровня. Другие встроенные средства включают в себя редактор форм, чтобы помочь создать графический интерфейс приложений, веб-редактор, класс дизайнера и проектировщика схемы базы данных. Visual Studio дает возможность создавать и подключать сторонние дополнения (плагины) для расширения функциональности почти на каждом уровне, в том числе добавление поддержки для систем управления исходным кодом (такие как, например, Subversion и Visual SourceSafe), добавляя новые наборы инструментов (например, для редактирования и визуального кода разработки объектно-ориентированных языков программирования) или инструментов для других аспектов процесса разработки программного обеспечения (например, клиент Team Explorer для работы с Team Foundation Server).

**1.4.2 C#** (произносится «си шарп») - объектно-ориентированный язык программирования. Разработан в 1998 - 2001 годах группой инженеров под руководством Андерса Хейлсберга в компании Microsoft как язык разработки приложений для платформы Microsoft .NET Framework и впоследствии был стандартизирован как ECMA-334 и ISO/IEC 23270.

C# входит в систему языков с C-подобным синтаксисом, где синтаксис его схож с C++ и Java. Язык владеет статической типизацией, поддерживает полиморфизм и перегрузку операторов. C#, опираясь на практику использования других языков, исключает некоторые модели, рекомендуемые себя как проблематичные при разработке программных систем, переняв многое от своих предшественников — языков C++, Pascal, Модула, Smalltalk и, в особенности, Java.

**1.4.3 Microsoft SQL Server** - система, разработанная корпорацией Microsoft для управления реляционными базами данных (РСУБД). Transact-SQL является основным используемым языком запросов, создан совместно с Microsoft и Sybase. Transact-SQL - это реализация стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Система используется для работы с базами данных размером от персональных до крупных БД масштаба с предприятие; конкурирует с другими СУБД в этом сегменте рынка.

**1.4.4 ASP.NET** (Active Server Pages для .NET) - это технология создания веб-приложений и веб-сервисов, разработанная компанией Майкрософт. **ASP.NET** является составной частью платформы Microsoft .NET и развитием более старой технологии Microsoft ASP. На данный момент последней версией этой технологии является **ASP.NET 5**.

ASP.NET во многом схож с более старой технологией ASP, что позволяет разработчикам легко ориентироваться и относительно перейти на ASP.NET. В то время как, внутреннее устройство ASP.NET колоссально отличается от ASP, так как она основана на платформе .NET и, соответственно, использует все новые возможности, предоставляемые этой платформой.

## 2 База знаний

База знаний является логической базой, содержащая правила логические вывода и изучения о человеческом опыте, знаниях в определенной предметной области. Основа познаний в самообучающихся системах охватывает информацию, которая является выводом решения прошлых задач.

Инновационная база знаний работает совместно с системой отслеживания и получения информации в виде текста или изображений. Для такого поиска потребуется некоторый тип классификации мнений и установленный формат представления. Онтологией называется иерархический метод представления знаний базового набора мнений и их отношений.

Вместе с тем, реальная основа познания (в отличие от общей базы данных) имеет не только практическую информацию, однако, и порядок вывода, обеспечивающий возможность работать, где автоматические выводы уже становятся доступными или снова вводимыми прецедентами, и являющимся наиболее для создания семантической (рациональный) отделкой сведения.

База знаний является важным компонентом интеллектуальной системы. Наиболее известным классом таких программ является экспертные системы. Они разработаны, чтобы найти пути решения проблем определенной предметной области, на основе записей БЗ и пользовательской описании ситуации.

Для создания экспертных систем хранения данных в организации может быть использовано простое знание: документация, учебные пособия, статьи по технической поддержке. Основной целью создания таких баз данных является помощь менее опытным людям найти уже существующее описание процесса решения любой задачи.

Два наиболее важных требования к информации, хранящейся в базе знаний интеллектуальной системы является:

1. Справедливость специфической и обобщенной информации, имеющейся в базе данных;
2. Актуальность информации, полученной с помощью базы знаний правил вывода.

Ниже перечислены некоторые из функций, которые могут (но не обязательно) быть в системе, которая управляет базами знаний.

- Автоматическое доказательство (вывод). Способность системы привносить новые знания от старого, чтобы найти шаблоны в базе знаний.

Принято считать, что база знаний отличается от базы данных только наличием механизма вывода.

- Заключение доказательства. Способность системы после выдачи ответа "объяснить" ход ее рассуждений, а именно «по первичному требованию».

- Самоанализ. Обнаружение противоречий, несоответствий в базе знаний, контроль за правильной организацией базы знаний.

- Машинное обучение. Превращение базы знаний в гибкую систему, адаптация к проблемной области. Аналогична человеческой способности «набраться опыта».

### **Классифицирование баз знаний**

В связи с разными видами уровней сложностей систем, в которых применяются базы знаний, различают:

- БЗ глобального масштаба (Интернет или Википедия)
- БЗ национальные (Википедия)
- БЗ отраслевые (Автомобильная энциклопедия)
- БЗ организаций
- БЗ экспертных систем
- БЗ специалистов

С точки зрения систем баз знаний в основе лежат принципы человеческого интеллекта. Интеллект владеет способностью подходить к решению любой проблемы с учетом накопленного опыта. По словам Хармона и Кинга (Harmon & King, 1985), Фишлера и Фиршейна (Fischler & Firschein, 1987) для человеческого интеллекта приемлемы следующие свойства:

- способность к обучению;
- способность выискивать аналогии;
- возможность создавать новые концепции, основанные на хорошо известных концепциях эффективной обработки спорных и несовместимых сообщений;
- возможность определить относительную важность различных компонентов проблемы;
- гибкий подход к решению проблемы;
- возможность разбить сложную проблему на её составные части;
- способность воспринимать мир моделирования;
- осознание и навыки использовать символические инструменты.

Машинное знание является аналогом искусственного интеллекта (ИИ). Основателем этой области является Алан Тьюринг, британский математик. Тем не менее, несмотря на то, что Тьюринг разработал оригинальную концепцию ИИ аж в 1937 году, официально ИИ появился только в 1956 году в Дартмутском колледже, во время встречи с группой ученых, чтобы обсудить потенциал компьютеров в области человеческого познания процесса

стимуляции. Один из организаторов конференции, Джон Маккарти предложил, терминов «искусственный интеллект».

ИИ является одна из отраслей науки. Это связано с компьютерами, стимулирующими процесс решения проблемы путем дублирования функций мозга человека. ИИ состоит из наборов аппаратных и программных средств, а также методов для моделирования деятельности человека как психического (мышления, принятия решений, рассуждений, решения проблем, обучения и поиска данных), так и физического (сенсорные и моторные навыки). Решения сложных проблем моделируется путем представления познавательного процесса человека и когнитивное моделирование решает проблемы, оценки знаний как личность.

Когнитивное моделирование и искусственный интеллект являются взаимосвязанными, но отличающимися дисциплинами. Когнитивное моделирование есть методика моделирования познавательного процесса человека, которая основана интеллектуальными рассуждениями и искусственного интеллекта - методика моделирования разумного поведения, в котором аргумент не является необходимым. Тем не менее, различия между этими двумя методами постепенно стираются.

Системы, основанные на знаниях, реализуются на основе этих интеллектуальных алгоритмов:

- нейронные сети;
- генетические алгоритмы;
- экспертные системы;
- нечеткая логика.

## **2.1. Парсинг программы**

Парсинг написан на технологии ASP.NET с использованием языка C#.

### **Инструменты:**

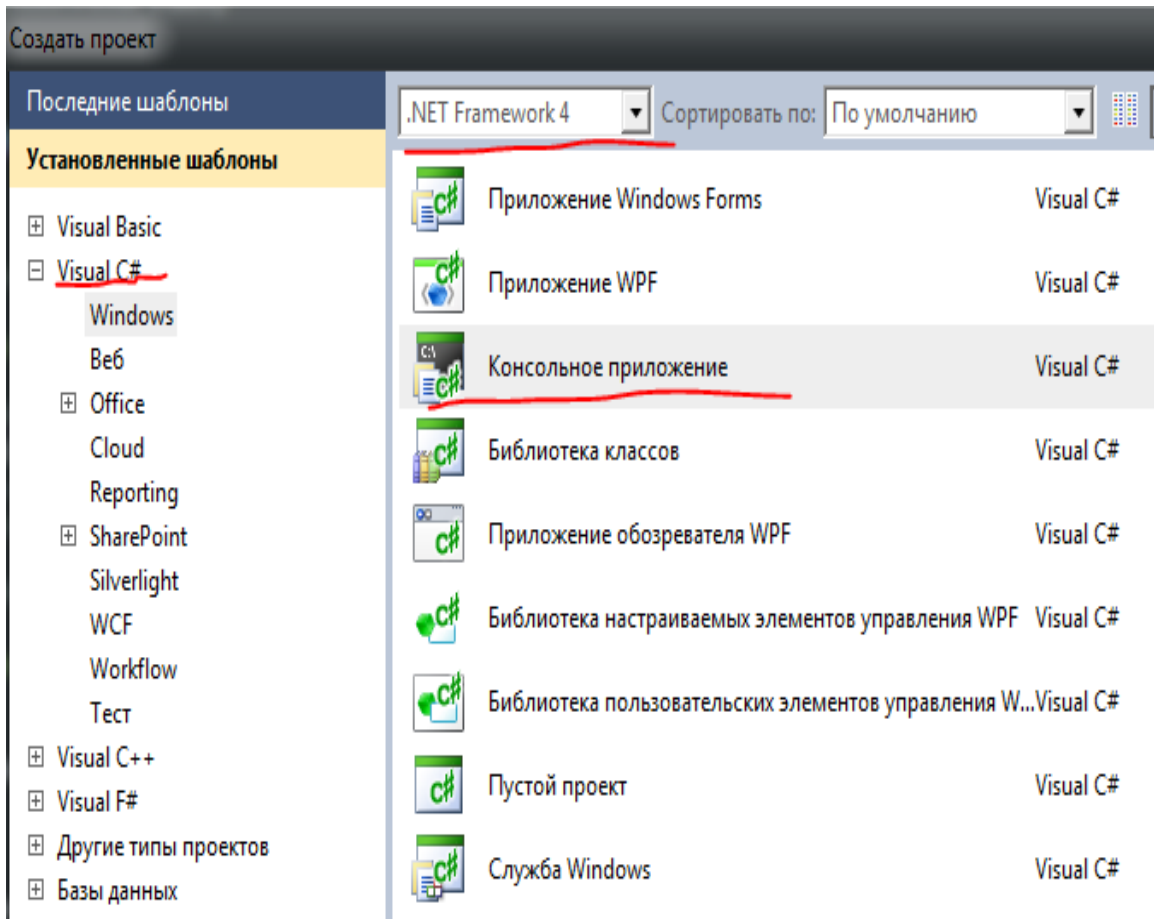
- Visual Studio 2015
- .Net Framework 4.0

В качестве парсера HTML используем HTML Agility Pack, но прежде чем парсить нужно сначала получить HTML, можно воспользоваться стандартным WebClient(), но далеко не секрет что сейчас все сайты усыпаны JavaScript который генерирует часть разметки, информации, а WebClient() к сожалению не может ждать пока отработает JS и только после этого забирать HTML. Значит нам он не подойдет. И на помощь приходит библиотека Watin!

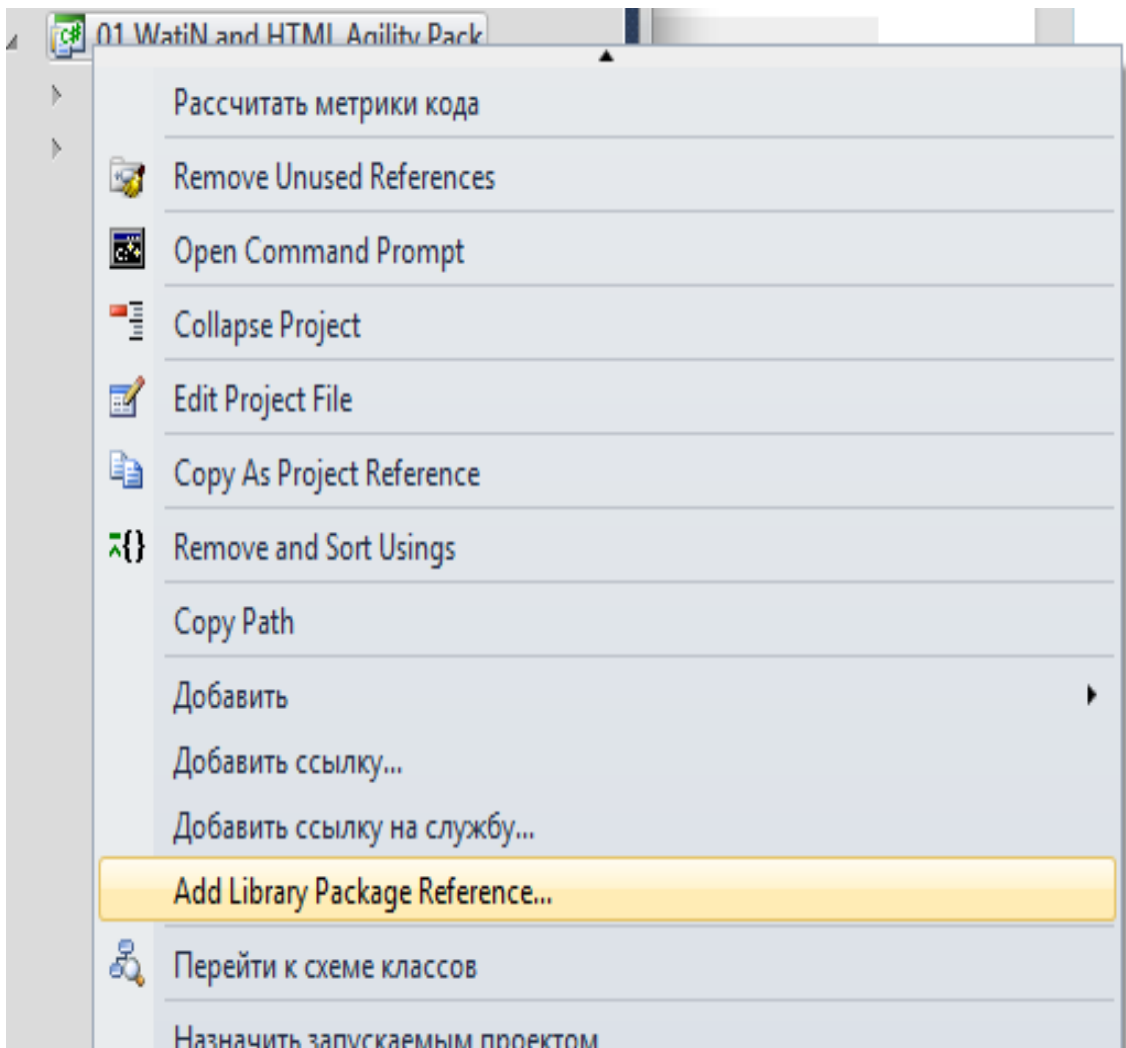
Watin - библиотека, которая помогает тестировать приложения. При программировании теста, и в реальном времени открывается браузер и выполняются различные нажатия кнопок ввод текст и т.д.

В Visual Studio создаем новое консольное приложение

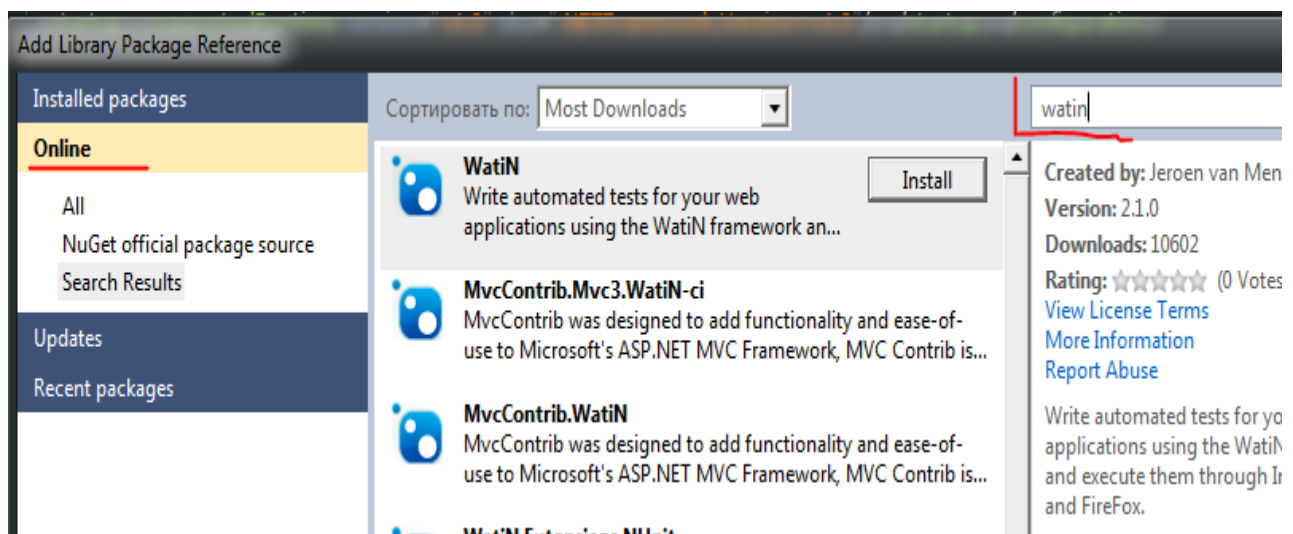




Далее устанавливаем нужные библиотеки

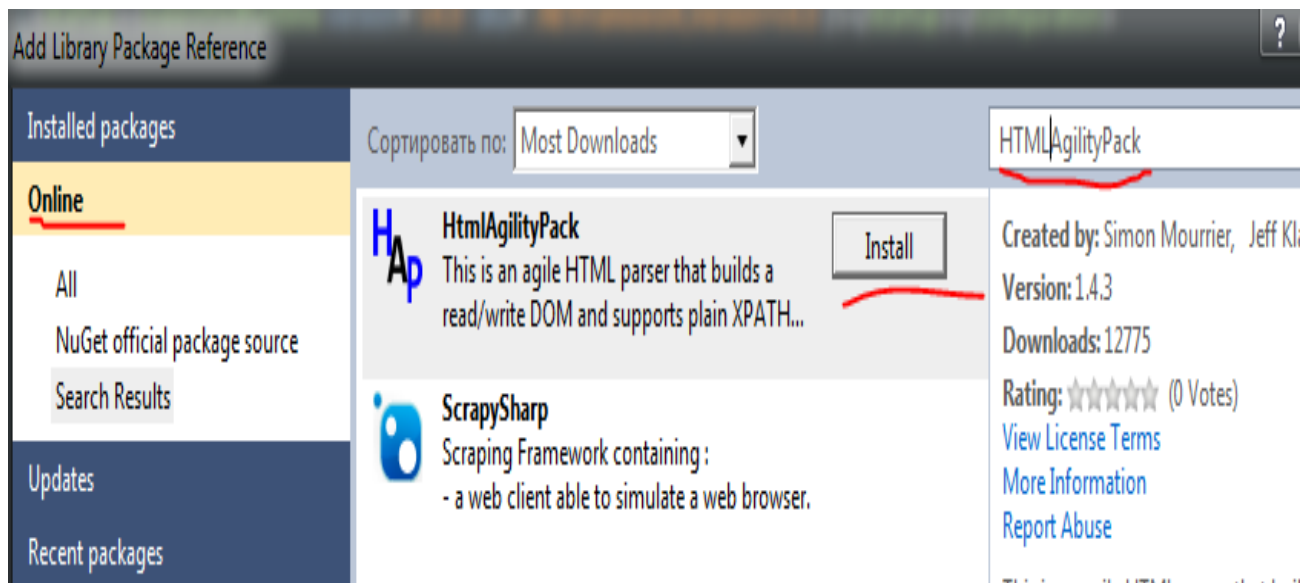


Открываем раздел "Online" и в строке поиска пишем "watin"



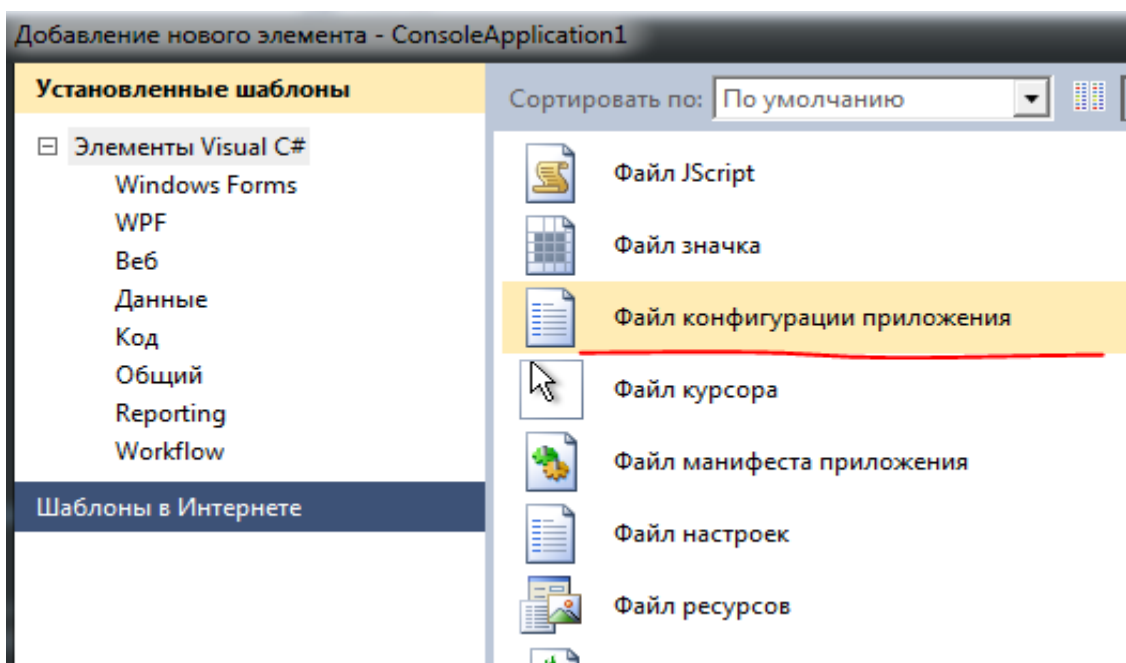
и жмем установить WatiN.

Далее пишем в поиске HTMLAgilityPack и тоже устанавливаем



Установка библиотек завершена.

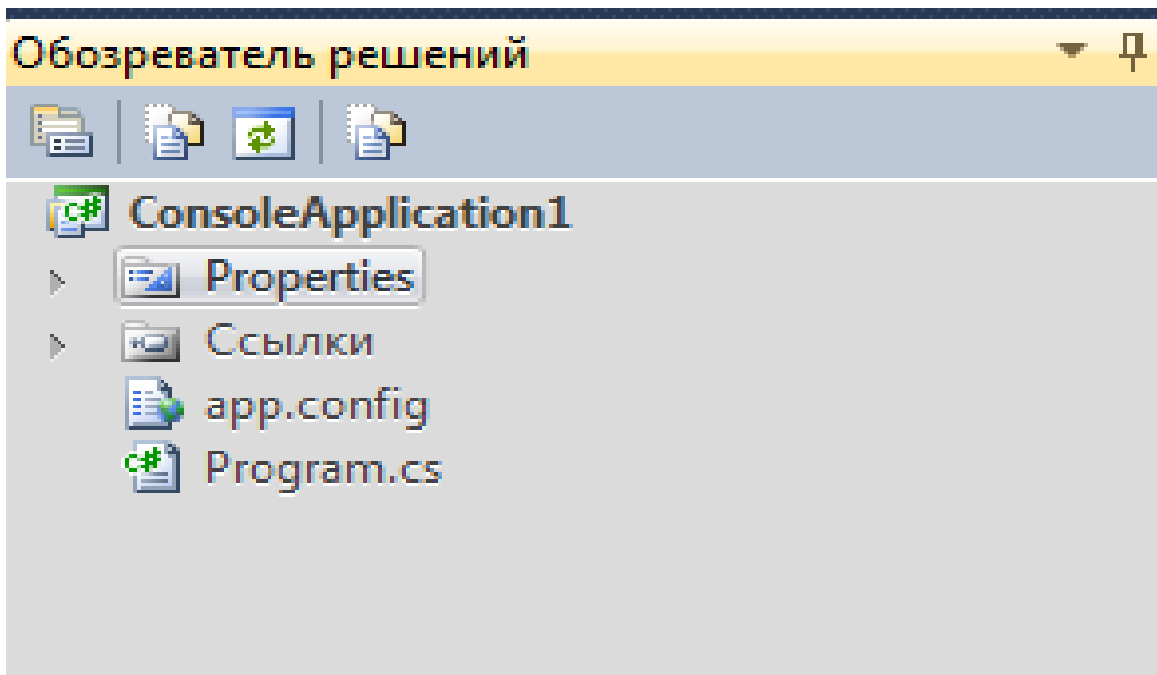
Далее добавляем файл конфигурации в наш проект с именем app.config



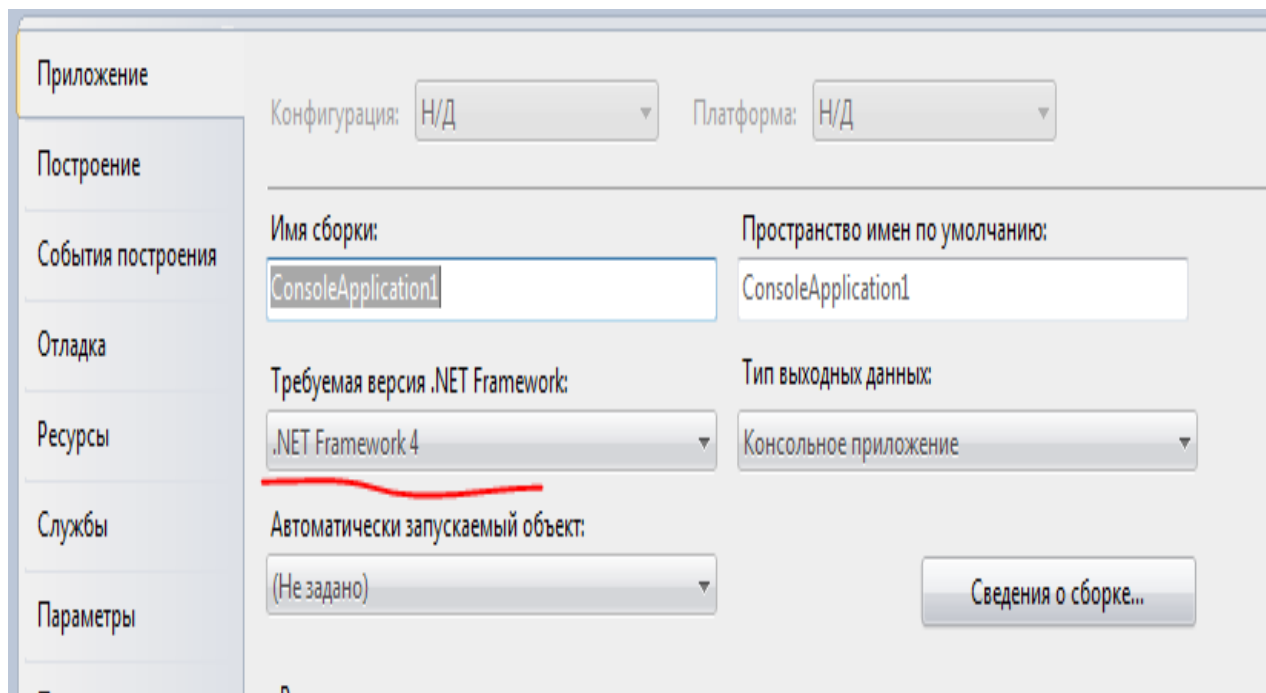
Во внутрь файла добавляем это:

1. `<?xml version="1.0"?>`
2. `<configuration>`
3. `<startup><supportedRuntime version="v4.0" sku=".NETFramework,Version=v4.0"/></startup></configuration>`

Далее изменим конфигурацию приложения, в solution нажмем на Properties два раза.



И изменим пункт "Требуемая версия .Net Framework" на ".Net Framework 4"



Далее открываем Program.cs и подключаем нужные библиотеки перед class Program{ }

1. using HtmlAgilityPack;
2. using System;
3. using WatiN.Core;

Перед static Main указываем, что это однопоточное приложение.

1. [STAThread]3

В итоге у вас должно получиться следующее

```
1.     using System.Text;
2.
3.
4.     namespace ConsoleApplication1
5.     {
6.         using HtmlAgilityPack;
7.         using System;
8.         using WatiN.Core;
9.
10.        class Program
11.        {
12.            [STAThread]
13.            static void Main()
14.            {
15.            }
16.        }
17.    }
```

\*namespace может отличаться  
далее в Main() {} указываем кодировку

1. Console.OutputEncoding = Encoding.UTF8;

и пишем структуру граббера html

1. using (var browser = new IE("http://www.yvision.kz"))
2. {
3. Settings.WaitForCompleteTimeOut = 999999999; /\*увеличиваем тайм-аут на всякий случай\*/
4. var doc = new HtmlDocument(); /\* инсталляция объекта парсера HTML Agility Pack\*/
5. doc.LoadHtml(browser.Body.OuterHtml); /\*помещаем в парсер полученный html с страницы yvision.kz\*/
6. }
7. Console.WriteLine("Done!"); /\*текст после завершения скрипта\*/

8. Console.ReadKey(); /\*ожидание нажатия клавиши\*/

Таким образом мы имеем следующий код:

1. скрипт открывает браузер IE
2. переходит по адресу <http://www.yvision.kz>
3. забирает HTML разметку
4. отдает парсеру

Далее нужно распарсить информацию с html и, например, отобразить их в консоле.

Смотрим структуру HTML, и видим что эти блоки лежат в div с классом .mainContent.main\_page.

Каждый отдельный блок имеет класс .half-1

Теперь можно запустить приложение и проверить - F5 в Visual Studio должно выглядеть примерно так:

```
<!-- end .article1-->pt--> }</a></li>
    <div class="hold"><a href="http://yvision.kz/post/242687"><
img src="http://in7.asset.yving.kz/post/short/242687.jpg?1332758555" alt=" " hei
    <div class="author"><a href="http://jolybolgish.yvision.kz"><
img src="http://yvision.kz/images/userpics/jolybolgish-small.jpg" alt=" " height
    <h3><a href="http://yvision.kz/post/242687"> Ғ Е Ғ
    <li><a class="comments" href="http://yvision.kz/post/242687#comment
    <li><a class="like " id="like_242687" post-id="242687" href="#">2</
    <li><a class="favorite " id="fave_242687" post-id="242687" href="#"
<!-- end .article1-->pt--> }</a></li>
    <div class="hold"><a href="http://yvision.kz/post/241939"><
img src="http://in7.asset.yving.kz/post/short/241939.jpg?1332537558" alt=" " hei
    <div class="author"><a href="http://zhumabekova.yvision.kz"><
img src="http://yvision.kz/images/userpics/zhumabekova-small.jpg" alt=" " height
    <h3><a href="http://yvision.kz/post/241939">Leadership Day Ғ
    <li><a class="comments" href="http://yvision.kz/post/241939#comment
    <li><a class="like " id="like_241939" post-id="241939" href="#">5</
    <li><a class="favorite " id="fave_241939" post-id="241939" href="#"
<!-- end .article1-->pt--> }</a></li>
    <div class="hold"><a href="http://yvision.kz/post/241586"><
img src="http://in5.asset.yving.kz/post/short/241586.jpg?1332339395" alt=" " hei
    <div class="author"><a href="http://arthalet.yvision.kz"><a href="http://yvision.kz/post/241586"> Ғ Ғ *
    <li><a class="comments" href="http://yvision.kz/post/241586#comment
    <li><a class="like " id="like_241586" post-id="241586" href="#">5</
    <li><a class="favorite " id="fave_241586" post-id="241586" href="#"
<!-- end .article1-->pt--> }</a></li>
    <div class="hold"><a href="http://yvision.kz/post/241273"><
img src="http://in9.asset.yving.kz/post/short/241273.jpg?1332245164" alt=" " hei
    <div class="author"><a href="http://kzcharityball.yvision.kz"
kzcharityball</a> / <span>20 Ғ В Ғ / 18:00</span> </di
    <h3><a href="http://yvision.kz/post/241273"> Т Ғ И Ғ И
    Ғ Ғ В Ғ Ғ Ғ Т Ғ Ғ И Ғ Ғ Ғ Ғ
    <li><a class="comments" href="http://yvision.kz/post/241273#comment
```

Блоки вывели, но нужно отсеять ненужную информацию, из каждого блока. Возьмем только ссылку на статью и адрес картинки. Видим что путь до ссылки "div.text > h3 > a" и путь до картинки у нас div.info> div.photo> div.hold> a > img

```

▼<div class="half-1">
  ▼<div class="article1">
    ▶<div class="text">...</div>
    ▼<div class="info">
      ▼<div class="photo">
        ▼<div class="hold">
          ▼<a href="http://yvision.kz/post/244394">
            
            "Появилась возможность задать вопрос Минюсту напрямую по поводу поправок"
          </a>
        </div>
      </div>
    </div>
    <!-- end .photo-->
    ▶<div class="author">...</div>
  </div>
  <!-- end .info-->

```

Запустим программу и увидим

```

Link item: http://yvision.kz/post/244394
Images patch: http://im4.asset.yvimg.kz/post/short/244394.jpg?
*****yvision.kz*****

Link item: http://yvision.kz/post/244394
Images patch: http://im4.asset.yvimg.kz/post/short/244394.jpg?
*****yvision.kz*****

Link item: http://yvision.kz/post/244394
Images patch: http://im4.asset.yvimg.kz/post/short/244394.jpg?
*****yvision.kz*****

Link item: http://yvision.kz/post/244394
Images patch: http://im4.asset.yvimg.kz/post/short/244394.jpg?
*****yvision.kz*****

Done!

```

## **3 Структура разработки**

### **3.1 Выбор и обоснование алгоритма программы**

Целью дипломной работы является создание информационной системы, которая оптимизирует работу медицинских учреждений. То есть облегчит прием пациентов, сократив огромные очереди, уменьшит время, затрачиваемое на осмотр пациента.

Структура БД является выгодным решением поставленной задачи, которая, в свою очередь, разбита на несколько взаимосвязанных таблиц, состоящих из определенных наборов полей.

### **3.2 Моделирование ИС**

Процесс создания и поддержания систем в виде жизненного цикла (ЖЦ) ИС описывает методологию проектирования информационных систем, представив его в виде последовательности этапов и процессов, запущенных на них. Определяется состав и последовательность выполняемых работ для каждого этапа, полученные результаты, методы и средства, необходимые для выполнения работ, ролей и обязанностей участников и т.д. Такое формальное описание жизненного цикла ИС позволяет планировать и организовывать процесс коллективного развития и управлять процессом.

Жизненный цикл ИС может быть представлен в виде последовательности событий, которые происходят с системой в процессе ее создания и использования. Модель жизненного цикла отражает различные состояния системы, с момента нашего создания в ИС необходимой и конечной точки его полного морального износа. Модель жизненного цикла есть структура, содержащая процессы, действия и задачи, которые выполняются в ходе разработки, эксплуатации и технического обслуживания программного обеспечения на протяжении всего срока службы системы, от определения требований до конца его использования.

Модель является не только полезным, но иногда и единственным инструментом исследования, при решении конкретной задачи, когда необходимо идентифицировать определенное свойство изучаемого объекта. Тот же объект может иметь множество моделей, а также различные объекты могут быть описаны с помощью одной модели.

Единая классификация моделей затруднена из-за многозначности понятия «модель» в науке и технике. Это может быть сделано по разным причинам: характер моделей и моделируемых объектов; в сферах применения и др.

**Моделированием** является представление объектной модели, чтобы получить информацию об этом путём проведения экспериментов с его моделью.



Термин "моделирование", как правило, воспринимается как процесс создания точного описания системы; метод обучения, который состоит в моделях создания и исследования.

Моделирование объекта облегчает исследование с целью создания, преобразования и дальнейшего развития. Оно используется для изучения существующей системы, когда реальный эксперимент проводить непрактично из-за значительных финансовых и трудовых затрат, а также когда необходимо провести анализ проектируемой системы, то есть которая физически не существует в организации.

Для формирования используются модели:

- блок-схема объекта;
- структурная и функциональная схема объекта;
- алгоритмы работы системы;
- аппаратная схема объекта;
- схема связи и другие.

Информационные модели используются для проектирования ИС, которые представляют объекты и процессы в виде чертежей, диаграмм, рисунков, таблиц, формул, текстов и т.д.

Информационной моделью называется модель объекта, процесса или явления, в котором представлены информационные аспекты моделируемого объекта, процесса или явления. Она является основой для разработки моделей ИС.

### **3.2.1 Моделирование БД**

Основные этапы, в котором проектируется база данных процесса информационной системы:

1. Концептуальное проектирование - сбор, анализ и редактирование требований к данным. С этой целью, проводятся следующие виды деятельности:

- обследование предметной области, изучение ее информационной структуры
- идентифицировать все фрагменты, каждый из которых характеризует представления пользователя, информационные объекты и связи между ними, процессы информационных объектов
- моделирование и интеграция всех представлений

В конце этого этапа мы получаем концептуальную модель, инвариантную к структуре базы данных. Она часто представлена как модель "сущность-связь".

2. Логическое проектирование - преобразование структуры данных требований к данным. Выход СУБД-ориентированной структуры спецификации баз данных и приложений. На данном этапе, часто моделируются базы данных для различных СУБД и проводится сравнительный анализ моделей.

3. Физическое проектирование определяет особенности хранения данных, методов доступа и т.д.

### **3.2.2 Диаграммное представление**

Следом кратко рассмотрим некоторые из особенностей одной из наиболее популярных семантических моделей данных - модель "сущность-связь" (часто называемой короткой ER-модели).

Об использовании разновидностей ER-модели основаны современные подходы к проектированию баз данных (в основном реляционного). Модель была предложена Ченом (Chen) в 1976 году. Моделирование домена основана на использовании графических диаграмм, в том числе небольшого числа разнородных компонентов. Из-за ясности представления концептуальных схем баз данных, ER-модели широко используются в CASE-системах, поддерживающих автоматизированное проектирование реляционных баз данных. Среди множества разновидностей ER-моделей одна из наиболее развитых используется CASE-системы в компании ORACLE. Точнее, мы сосредоточимся на структурной части модели.

### **3.2.3 Основные понятия модели Entity-Relationship (Сущность-Связи)**

Сущностью является реальный или представляемый объекты, информация о которых должна быть сохранена и быть в наличии. Сущность представляется в виде прямоугольника, содержащего имя сущности в диаграммах ER-модели. Имя сущности есть имя типа. Имя сущности может сопровождаться примерами конкретных объектов этого типа.

При определении типа объекта вы должны убедиться, что каждый экземпляр сущности может быть неотличим от любого другого экземпляра той же сущности. Это требование во многом схож с отсутствием спроса дубликатов кортежей в реляционных таблицах.

Связь - это графически изображенная связь между двумя типами сущностей. Как и сущность, связь - это типичная концепция, все экземпляры обоих связываемых типов сущностей подчиняются правилам, установленным связью. В обсуждаемом здесь варианте ER-модели, эта ассоциация всегда бинарна и может существовать между двумя разными типами структур или между типом сущности и им же самим (рекурсивное отношение). В любом отношении есть два конца (в соответствии с существующей парой связываемых сущностей), каждый из которых определяет конец имени товарищества, степень конца соединения (сколько экземпляров типа сущности должны присутствовать в каждом случае этого тип связи), обязательность связи (т.е. есть ли какие-либо сущности из этого типа экземпляра, которые должны участвовать в определенном экземпляре этого вида связи).

Связь представлена в виде ненаправленной линии, соединяющей две сущности или ведущей от сущности к себе. На месте "стыковки" связи с сущностью используются:

1. трехточечный вход в прямоугольник сущности, если для этой сущности в связи могут (или должны) использоваться много экземпляров сущности;

2. одноточечный вход, если в связи может (или должен) участвовать только один экземпляр сущности.

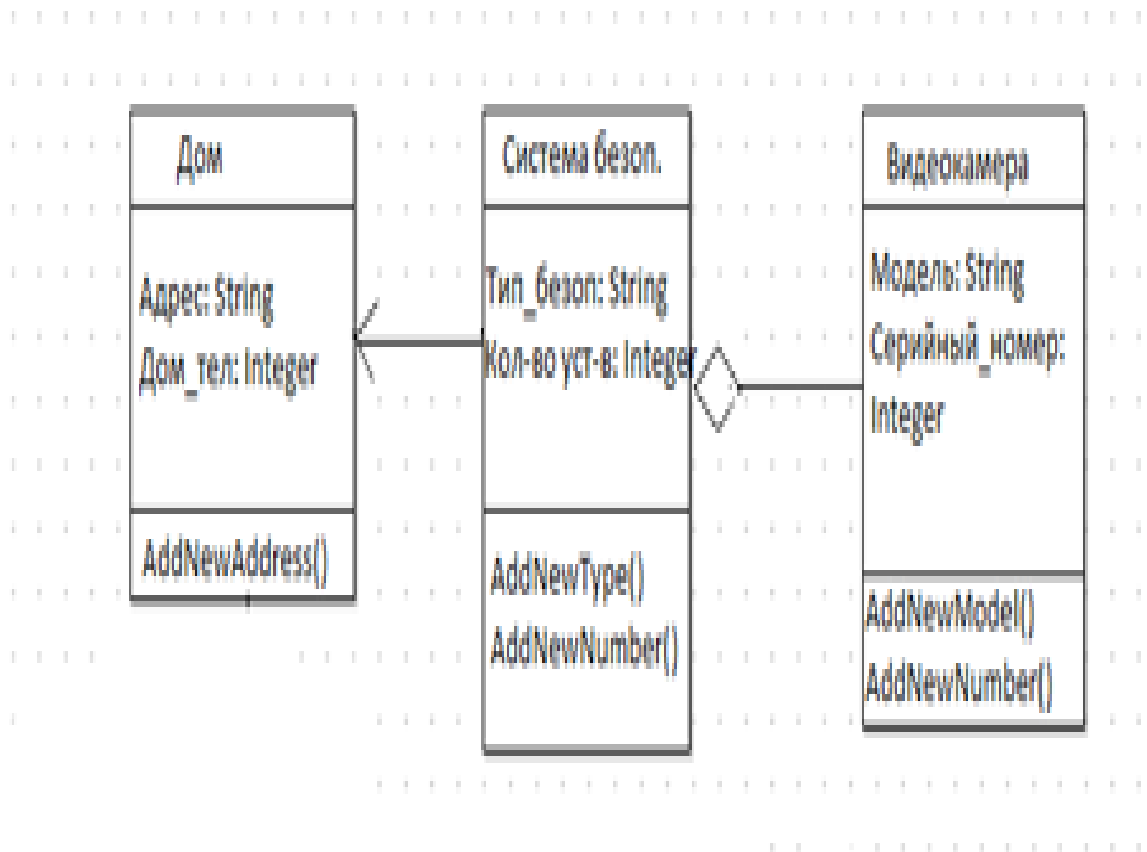
Обязательный конец связи изображается сплошной линией, а необязательный – прерывистой линией.

### 3.2.4 Диаграммы

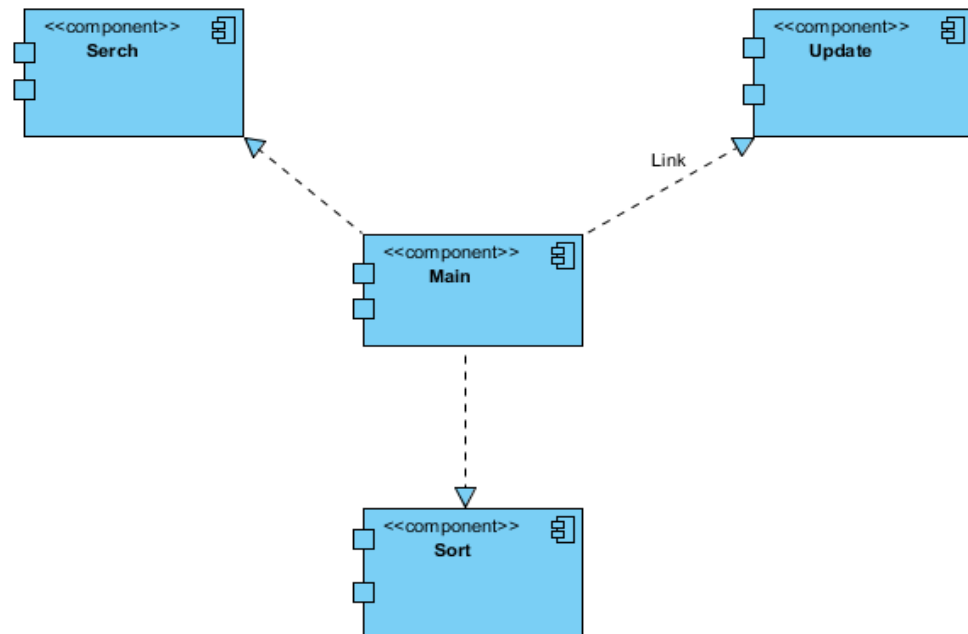
*Диаграмма* — графическое представление множества элементов, наиболее часто изображается как связный граф из вершин (предметов) и дуг (отношений). UML включает 8 видов диаграмм:

- 1) диаграммы классов (Class diagrams);
- 2) диаграммы прецедентов (Use Case diagrams);
- 3) диаграммы последовательностей (Sequence diagrams);
- 4) диаграммы кооперации (Collaboration diagrams);
- 5) диаграммы состояний (State diagrams);
- 6) диаграммы действий (Activity diagrams);
- 7) диаграммы компонентов (Component diagrams);
- 8) диаграммы развертывания (Deployment diagrams).

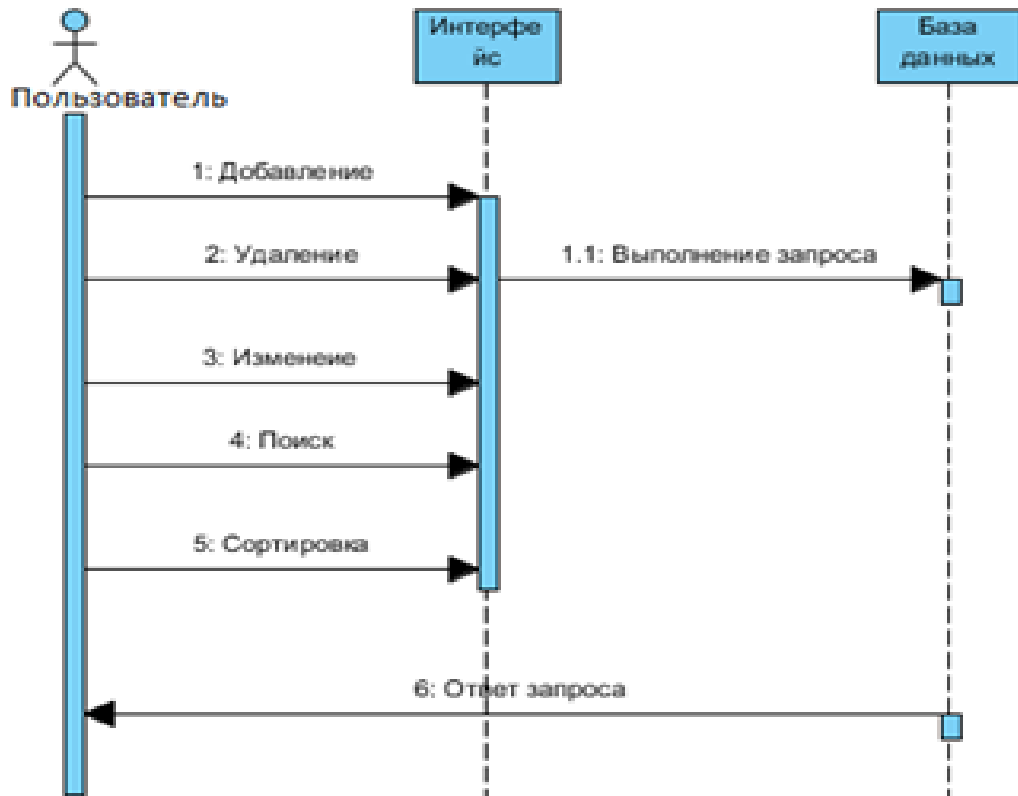
#### Диаграмма классов:



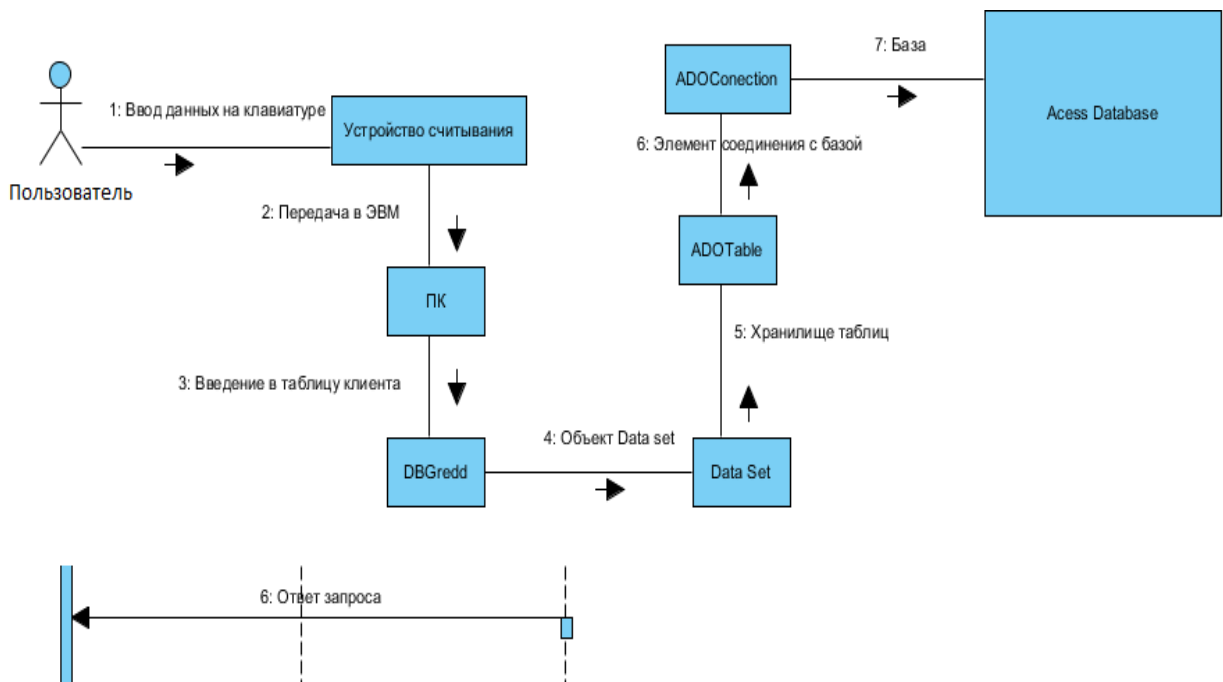
**Диаграмма компонентов:**



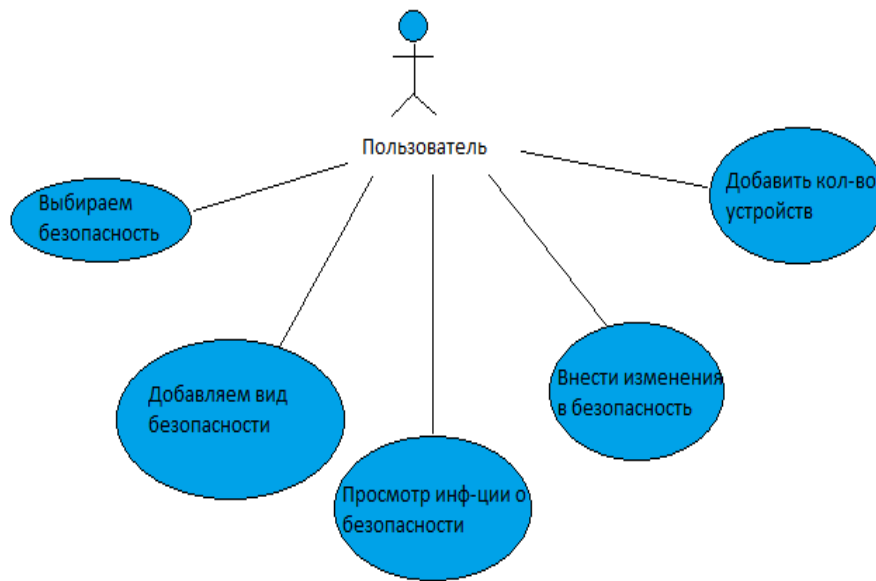
**Диаграмма последовательности:**



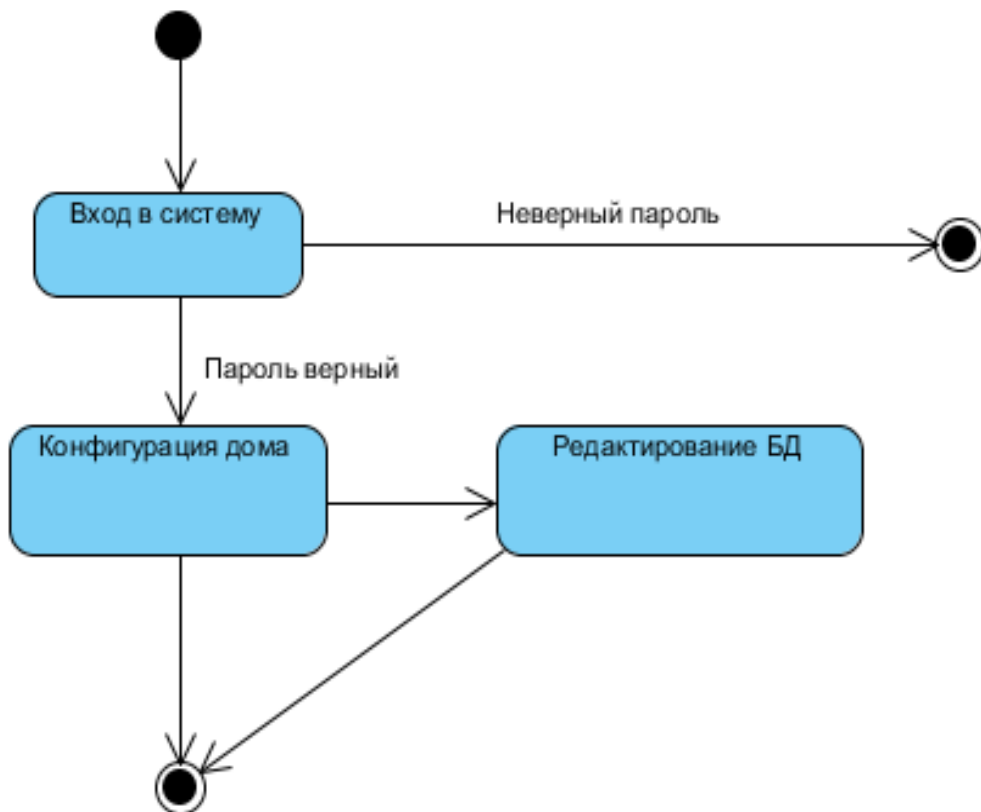
### Диаграмма коопераций:



**Диаграмма прецедентов:**



**Диаграмма состояний:**

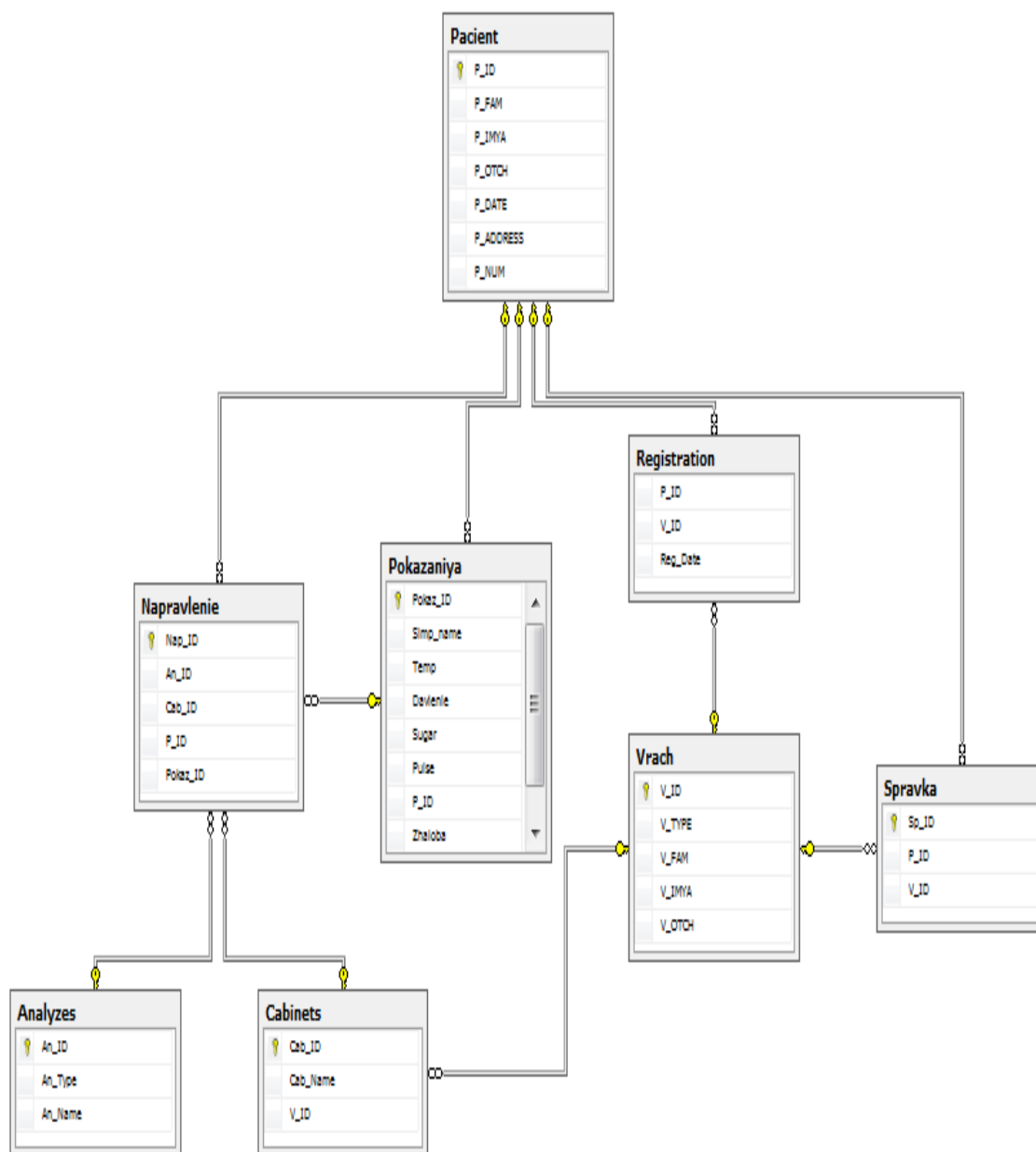


**Диаграмма развертывания:**



### 3.3 Реализация программы

#### Схема БД программы



Первым делом нужно авторизоваться. В данной программе 3 вида пользователей: специалист, лаборант, врач.

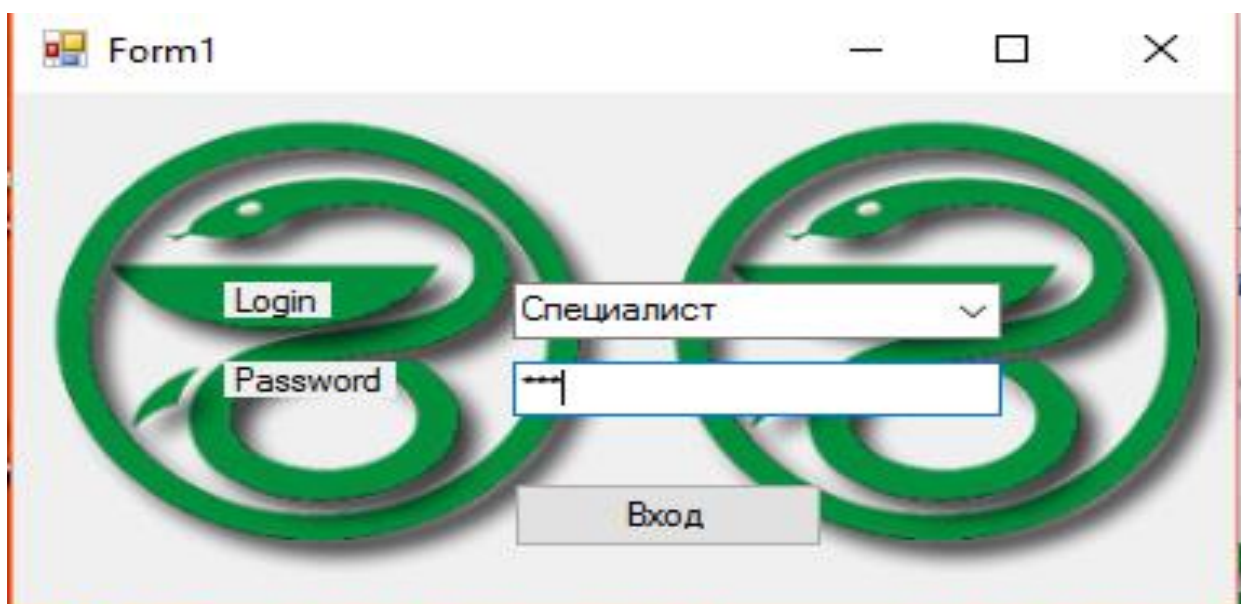
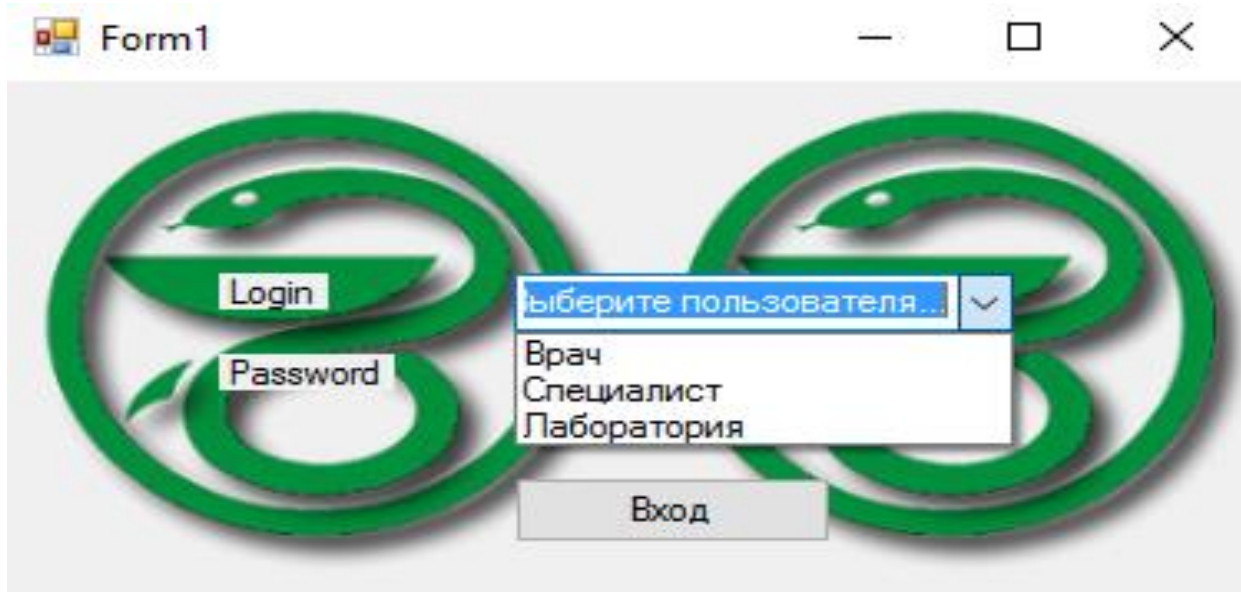
Специалист сидит в кабинете первичного осмотра, принимает пациентов, регистрирует их и заносит информацию в систему.

Лаборант принимает заявки на анализы, затем полученные анализы вносит в базу.

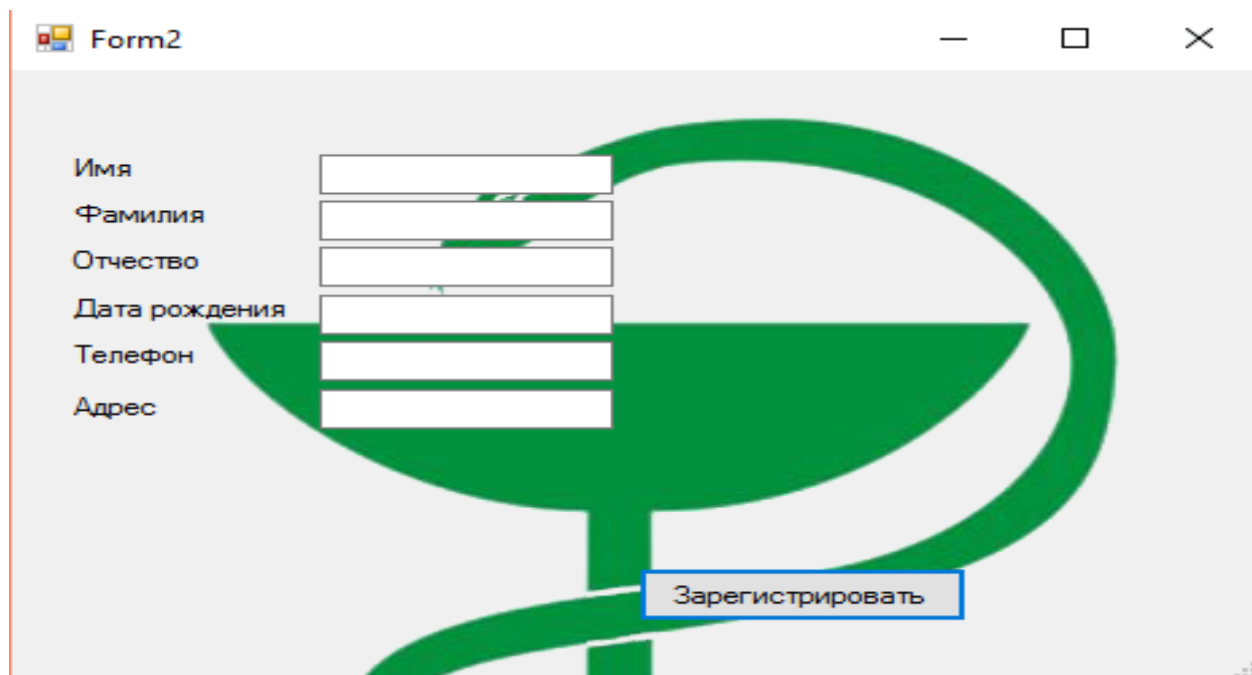
Врач принимает пациентов уже на вторичный прием. Имея на руках уже готовые результаты анализов, ставит окончательный диагноз и назначает лечение.



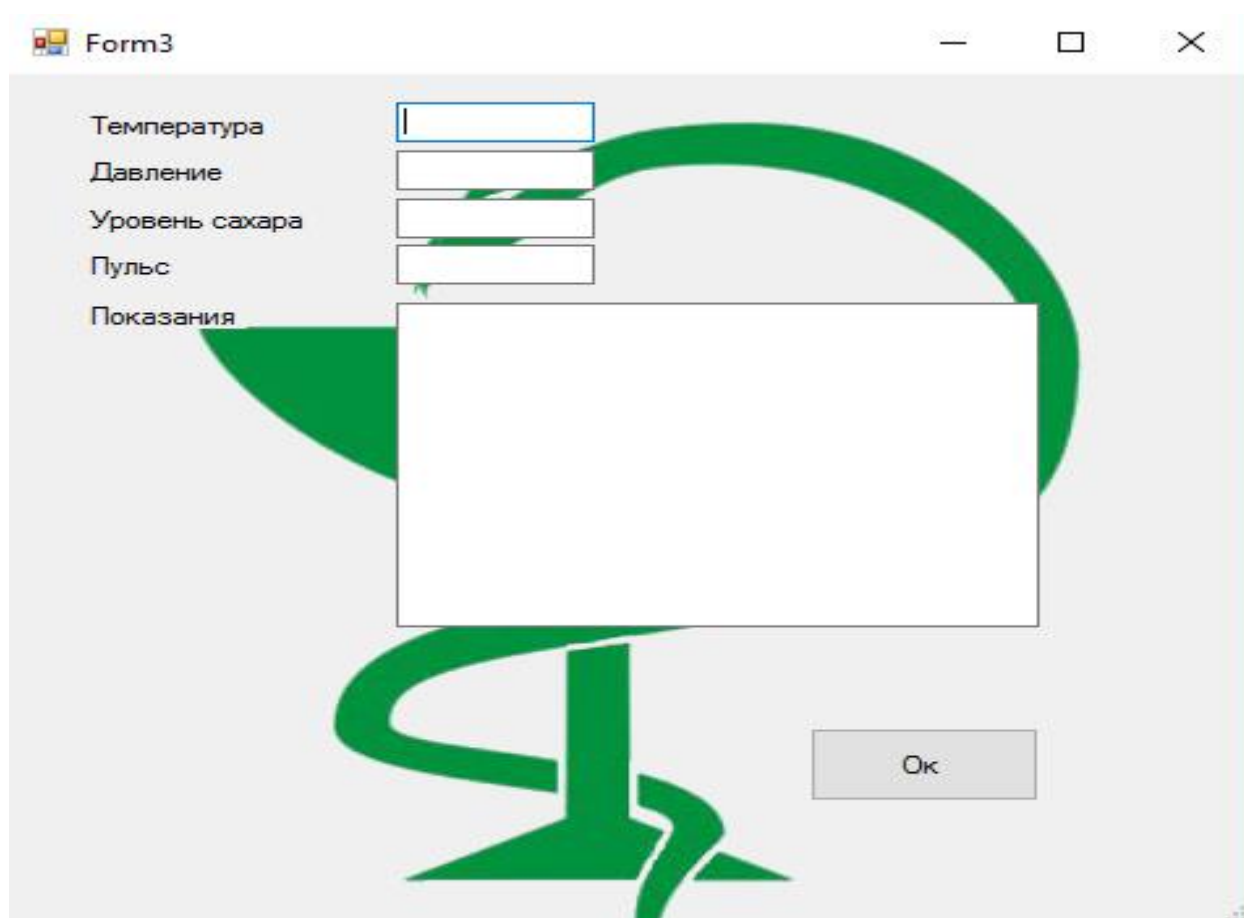
Выбираем пользователя Специалист, вводим пароль и входим в систему.



Специалисту, чтобы зарегистрировать пациента, нужно заполнить поля, затем нажать на кнопку «Зарегистрировать».



The screenshot shows a window titled "Form2" with a light gray background and a large green stylized logo. On the left, there are labels for "Имя", "Фамилия", "Отчество", "Дата рождения", "Телефон", and "Адрес". To the right of each label is a white rectangular input field. At the bottom right, there is a button labeled "Зарегистрировать" with a blue border.



The screenshot shows a window titled "Form3" with a light gray background and the same green stylized logo. On the left, there are labels for "Температура", "Давление", "Уровень сахара", "Пульс", and "Показания". To the right of each label is a white rectangular input field. The "Показания" field is significantly larger than the others. At the bottom right, there is a button labeled "Ок".

После регистрации пациента, специалист производит первичный осмотр, заносит результаты и показания в базу. После нажатия кнопки «Ок», программа выдает направление со списком нужных анализов, которые нужно сдать пациенту.

Form6

Направление №

ФИО

Список анализов

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

Место для печати

Распечатать

Пользователь «Врач», имея на руках полную информацию о пациенте, может произвести дополнительный осмотр, вынести окончательный диагноз и назначить лечение, распечатать справку и лист назначения.

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:**

## **4 Техничко–экономическое обоснование**

### **4.1 Описание работы и обоснование необходимости**

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

Данный проект ориентирован на улучшение работ медицинских учреждений, оказывающих первичную медико-санитарную помощь.

Цель работы - создание информационной системы, которая оптимизирует работу медицинских учреждений. То есть облегчит прием пациентов, сократив огромные очереди, уменьшит время, затрачиваемое на осмотр пациента.

Три важных составляющих в работе современного медицинского учреждения это учет, контроль и управление. Автоматизация медицинских учреждений создает единую и надежную систему.

Ввиду того, что на сегодняшний день процесс информатизации затронул, скажем так, все сферы деятельности человека, то и у передовой части медицинского менеджмента сейчас весьма актуальным вопросом является: можно ли улучшить работу медицинских учреждений, если применить в них современные IT - технологии? И если да, то каким образом?

Для начала, нужно принять во внимание то, что система - это комплексное решение для применения IT в медицинских учреждениях. Сама система же разработана для модернизации работы учреждения в целом и охватывает все важные структуры и процессы.

Это, в частности:

- Хранение информации в компактном электронном виде
- Ведение электронной истории болезней
- Выдача необходимого количества анализов
- Возможность лаборатории принимать заявки на анализы и передавать результаты в общий электронный банк данных
- Выдача справки с листом назначения

### **4.2 Трудовые ресурсы, используемые в работе.**

В ходе разработки используется интеллектуальный труд, стоимость затрат которого выше, чем физического труда.

В разработке приняли участие:

- дизайнер - разработка интерфейса;
- программист - разработка алгоритмов и программирование;

Количество сотрудников, задействованных в разработке представлено в таблице 4.1.

Таблица 4.1 – Сотрудники и их заработная плата

<b>Должность</b>	<b>Количество человек</b>	<b>Зарботная плата в месяц, тенге</b>
Программист	1	100 000,00
Дизайнер	1	70 000,00
<b>Итого</b>	<b>2</b>	<b>170 000,00</b>

#### 4.3 Оборудование, используемое в работе

Оборудование, используемое при разработке программного обеспечения представлено в таблице 4.2.

Таблица 4.2 – Перечень оборудования, необходимого для разработки

<b>Наименование изделий</b>	<b>Характеристика</b>	<b>Количество единиц</b>	<b>Цена за единицу, тенге</b>	<b>Общая сумма, тенге</b>
Ноутбук	Intel Core i7-5500U 6Gb DDR3 GeForce GT 940M 2GB HDD 1000Gb 15.6" LED LCD (16:9)	2	75 000,00	150 000,00
Многофункциональное устройство	Samsung SL-M2070	1	30 000,00	30 000,00
<b>Итого</b>				<b>180 000,00</b>

Цены на оборудование представлены без учета НДС.

#### 4.4 Программное обеспечение, используемое в работе

При разработке ПО было использовано следующее программное обеспечение:

- Windows 8 – операционная система;
- Microsoft SQL Server 2014 – PCСУБД;
- Visual Studio 2015 – интегрированная среда разработки.

Программное обеспечение, использованное при разработке ПО, представлено в таблице 4.3.

Таблица 4.3 – перечень программного обеспечения, необходимого для разработки ПО медицинских учреждений

<b>Программное обеспечение</b>	<b>Стоимость, тенге</b>
Windows 8 максимальная	20 000,00
Microsoft SQL Server 2014	бесплатно
Visual Studio 2015	бесплатно
<b>Итого</b>	<b>20 000,00</b>

Цены на ПО приведены без учета НДС.

#### 4.5 Сроки реализации проекта

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

- Исследование модели предметной области;
- Разработка схемы данных;
- Проработка структуры алгоритма работы системы;
- Проектирование пользовательского интерфейса для системы;
- Тестирование программного обеспечения;
- Оформление отчетов.

Этапы и сроки реализации проекта отображены в таблице 4.4.

Таблица 4.4 - Этапы и сроки реализации проекта

<b>Перечень работ</b>		<b>Недели от начала работ</b>									
		1	2	3	4	5	6	7	8	9	10
1 этап	Исследование модели предметной области										
	Подбор и изучение литературы										
	Анализы и статистика										
	Опрос среди медицинских учреждений										
2 этап	Проектирование и создание базы данных										
	Заполнение базы данных										
	Создание интерфейса										

3 этап	ПО											
4 этап	Проработка структуры алгоритма											
	Программирование ПО											
5 этап	Разработка админ. панели											
	Тестирование ПО											
6 этап	Отладка ПО											
	Оформление НИР											
	Проверка и сдача отчета											

#### 4.6 Расчет стоимости работы по разработке

Расчет стоимости работы по разработке является наиважнейшей частью экономического анализа, ибо на основе данного расчета определяются затраты рабочего времени на разработку проекта на каждом этапе, а также трудовые затраты.

Затраты на разработку данного проекта определяются по формуле

$$C = \Phi OT + O_c + A + \Xi + C_{\text{пр}} + H \quad (4.1)$$

где,  $\Phi OT$  - фонд оплаты труда;

$O_c$  - социальный налог;

$A$  - амортизационные отчисления;

$\Xi$  - затраты на электроэнергию;

$C_{\text{пр}}$  - прочие расходы;

$H$  - накладные расходы.

##### 4.6.1 Расчет затрат на оплату труда

Затраты на оплату труда сотрудников, задействованных в разработке проекта, рассчитываются по формуле:

$$\Phi OT = Z_{\text{осн}} + Z_{\text{доп}} \quad (4.2)$$

где,  $Z_{\text{осн}}$  - основная заработная плата;

$Z_{\text{доп}}$  - дополнительная заработная плата.

Труд программиста и дизайнера принят условно, на договорной основе в размере 100 000,00 и 70 000,00 тенге соответственно.

В зависимости от этапов разработки программного обеспечения участники были задействованы неравноценно. Чтобы определить



вкладываемый труд, нужно рассчитать средний дневной заработок, а затем и общий размер заработной платы, в зависимости от их фактического участия.

Средний дневной заработок каждого работника рассчитывается по формуле:

$$D = \frac{ЗП_m}{Д_r} \quad (4.3)$$

где  $ЗП_m$  – ежемесячный размер заработной платы;

$Д_r$  – количество рабочих дней в месяце (это 22 дня – шестидневная рабочая неделя).

1) Для программиста

$$D = \frac{100000}{22} = 4545,45 \text{ тенге/день}$$

2) Для дизайнера

$$D = \frac{70000}{22} = 3181,82 \text{ тенге/день}$$

Заработная плата за один час вычисляется по формуле:

$$H = \frac{ЗП_m}{Д_r * Ч_r}$$

(4.4)

где,  $ЗП_m$  – ежемесячный размер заработной платы;

$Д_r$  – количество рабочих дней в месяце.

$Ч_r$  – Количество часов рабочего дня ( $Ч_r = 8$ )

1) Для программиста

$$H = \frac{100000}{22 * 8} = 568,18 \text{ тенге/час}$$

2) Для дизайнера

$$H = \frac{70000}{22 * 8} = 397,73 \text{ тенге/час}$$

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

$$t_n = \frac{T}{q_n * z * K} \quad (4.5)$$

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.

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

Для каждого вида работ произведены аналогичные расчеты.

$$t_n = 3 + 4 + 3 + 4 + 6 + 3 + 4 + 8 + 13 + 6 + 2 + 2 + 1 + 1 = 60 \text{ дней}$$

Таким образом, для проведения всех работ необходимо 60 дней.

Сводные данные по расчету заработной платы персонала, задействованного в разработке проекта приведены в таблице 4.5.

Таблица 4.5 – Сводные данные по расчету основной заработной платы персонала задействованного в разработке проекта.

Наименование этапов	Исполнитель	Трудоемкость, норма-час	Длительность цикла, дни	Заработная плата за час работы, тенге	Сумма заработной платы, тенге
Исследование модели предметной области	Программист	24	3	568,18	13 636,32
Разработка схемы дизайна ПО	Дизайнер	32	4	397,73	12 727,36
Подбор и изучение литературы	Программист	24	3	568,18	13 636,32

Анализы и статистика	Программист	32	4	568,18	18 181,76
Проектирование и создание базы данных	Программист	48	6	568,18	27 272,64
Заполнение базы данных	Программист	24	3	568,18	13 636,32
Создание интерфейса ПО	Дизайнер	32	4	397,73	12 727,36
Верстка	Программист	64	8	568,18	36 363,52
Программирование ПО	Программист	104	13	568,18	59 090,72
Разработка админ. панели	Программист	48	6	568,18	27 272,64
Тестирование ПО	Программист	16	2	568,18	9 090,88
Отладка ПО	Программист	16	2	568,18	9 090,88
Оформление НИР	Программист	8	1	568,18	4 545,44
Проверка и сдача отчета	Программист	8	1	568,18	4 545,44
<b>Итого</b>		<b>480</b>	<b>60</b>	<b>7 613,62</b>	<b>26 1818,00</b>

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

$$Z_{\text{доп}} = Z_{\text{осн}} * 0,1 \quad (4.6)$$

и составит

$$Z_{\text{доп}} = 261818 * 0,1 = 26182 \text{ тенге};$$

Отсюда следует, что затраты на оплату труда согласно произведенным расчетам и в соответствии с формулой 4.2 равна

$$\text{ФОТ} = 261818 + 26182 = 288000 \text{ тенге}$$

#### 4.6.2 Расчет затрат по социальному налогу

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

$$O_c = (\text{ФОТ} - \text{ПО}) * 11\% \quad (4.7)$$

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

$$\begin{aligned} \text{ПО} &= \text{ФОТ} * 10\% \\ \text{ПО} &= 288000 * 0,1 = 28800 \text{ тенге.} \end{aligned} \quad (4.8)$$

Таким образом, в соответствии с произведенными расчетами и согласно формуле 4.7 размер отчислений на социальные нужды равна

$$O_c = (288000 - 28800) * 0,11 = 28512 \text{ тенге}$$

#### 4.6.3 Расчет амортизационных отчислений

Амортизационные отчисления рассчитываются по формуле

$$A_i = \frac{H_A * C_{\text{ПЕР}} * N}{100 * n} \quad (4.9)$$

где,  $H_A$  – норма амортизации;

$C_{\text{ПЕР}}$  – первоначальная стоимость оборудования;

$N$  – количество дней на выполнение работ;

$n$  – количество рабочих дней в году.

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

Следовательно, амортизационные отчисления по используемому оборудованию, в соответствии с формулой 4.9 равны

- на ноутбук

$$A_1 = \frac{40 * 75000 * 2 * 60}{100 * 365} = 9863,01 \text{ тенге}$$

- на многофункциональное устройство

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

- на программное обеспечение

$$A_3 = \frac{15 * 20000 * 60}{100 * 365} = 493,15 \text{ тенге}$$

$$A_i = 9863,01 + 131,5 + 493,15 = 10487,66 \text{ тенге}$$

Сводные результаты расчета амортизационных отчислений представлены в таблице 4.6.

Таблица 4.6 – Сводные данные по расчету затрат на амортизацию

Наименование оборудования	Количество	Норма амортизации, %	Сумма амортизации, тенге
Ноутбук	2	40	9 863,01
Многофункциональное устройство	1	40	131,5
Программное обеспечение	-	15	493,15
<b>Итого</b>			<b>10 487,66</b>

#### 4.6.4 Расчет затрат на электроэнергию

Ввиду того, что в процессе разработки используется электрооборудование, надлежит рассчитать затраты на электроэнергию. Затраты на электроэнергию для производственных нужд включают в себя расходы электроэнергии на оборудование и дополнительные нужды, рассчитываются по формуле

$$\mathcal{E} = \mathcal{Z}_{\text{эл.эн.об.}} + \mathcal{Z}_{\text{доп.}} \quad (4.10)$$

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

$$\mathcal{Z}_{\text{эл.эн.об.}} = W * T * S * K_{\text{исп}} \quad (4.11)$$

где,  $W$  – потребляемая мощность, Вт;

$T$  – время работы, часы;

$S$  – тариф (1кВт = 16,02 тенге);

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

$$\mathcal{Z}_{\text{эл.эн.об. (ноутбук)}} = 0,7 * 480 * 16,02 * 0,9 * 2 = 9688,9 \text{ тенге}$$

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

Общая сумма затрат на электроэнергию основного оборудования согласно формуле 4.11 равна

$$Z_{\text{эл.эн.об.}} = 9688,9 + 415,24 = 10104,14 \text{ тенге}$$

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

$$Z_{\text{доп}} = Z_{\text{эл.эн.об.}} * 5\% \quad (4.12)$$

и равны

$$Z_{\text{доп}} = 10104,14 * 0,05 = 505,2 \text{ тенге}$$

Следовательно, суммарные затраты на электроэнергию, согласно формуле 4.11 равны

$$\Xi = 10104,14 + 505,2 = 10609,34 \text{ тенге}$$

Сводные результаты расчета затрат на электроэнергию представлены в таблице 4.7

Таблица 4.7 – 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	60	0,9	480	10 104,14
МФУ	1	0,9	4	0,9	32	505,2
<b>Итого</b>						<b>10 609,34</b>

#### 4.6.5 Расчет накладных и прочих расходов

Прочие расходы включают в себя:

1) расходы на интернет на 60 дней (3 месяца). Стоимость интернета за один месяц составляет 3800 тенге. Получаем:

$$3800 * 3 = 11400 \text{ тенге}$$

2) расходы на канцелярские товары:

- упаковка бумаги формата А4 – 1000 тенге;
- упаковка ручек – 300 тенге;
- 2 карандаша – 100 тенге;
- ластик – 50 тенге;
- линейка - 100 тенге;

- стикеры – 200 тенге.

Общая стоимость канцелярских товаров

$1000 + 300 + 100 + 50 + 100 + 200 = 1750$  тенге.

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

Площадь помещения:

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

Стоимость одного квадратного метра равна 1500 тенге. Получаем:

$12 * 1500 * 3 = 54000$  тенге.

Прочие расходы равны

$C_{\text{ПР}} = 11400 + 1750 + 54000 = 67150$  тенге.

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

$$H = (\text{ФОТ} + O_c + A + \text{Э} + C_{\text{ПР}}) * 50\% \quad (4.13)$$

$H = (288000 + 28512 + 10487,66 + 10609,34 + 67150) * 0,5 = 202\,379,5$   
тенге

#### 4.6.6 Расчет стоимости по всем статьям затрат и определение структуры затрат

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

$$C = 288000 + 28512 + 10487,66 + 10609,34 + 67150 + 202379,5 = \\ = 607\,139,00 \text{ тенге}$$

Смета затрат по разработке ПО медицинских учреждений, а также структура расходов представлены в таблице 4.8 и на рисунке 4.1

Таблица 4.8 – Суммарные данные по стоимости разработки проекта

Наименование статьи	Сумма, тенге	%
<b>затрат</b>		
Фонд оплаты труда	288000	47
Социальный налог	28512	5
Амортизация	10487,66	2
Затраты на электроэнергию	10609,34	2
Прочие расходы	67150	11
Накладные расходы	202379,5	33
<b>Итого</b>	<b>607139</b>	<b>100</b>

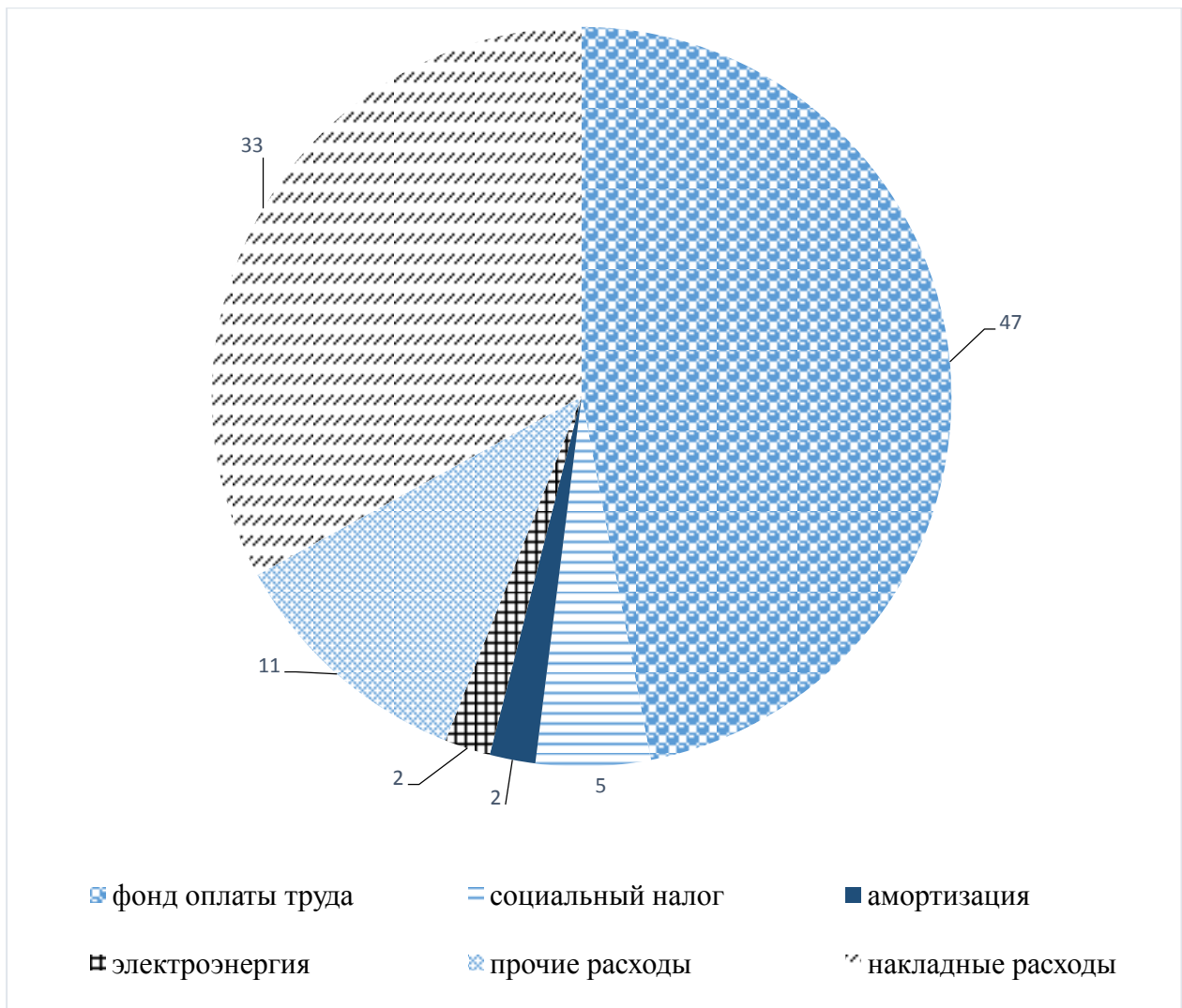


Рисунок 4.1 – Структура затрат по разработке сайта

#### 4.7 Цена интеллектуального руда

Цена реализации проекта складывается из стоимости и чистого дохода

$$Ц = С + П \quad (4.14)$$

где, С – стоимость продукта;

П – чистый доход.

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

$$Ц_{П} = С * \left(1 + \frac{Р}{100}\right) \quad (4.15)$$

где, Р – рентабельность.

$$Ц_{П} = 607139 * \left(1 + \frac{20}{100}\right) = 728567 \text{ тенге}$$



Цена реализации проекта рассчитывается по формуле

$$C_p = C_{\Pi} + \text{НДС} \quad (4.16)$$

где, НДС – налог на добавленную стоимость по ставке 12%

$$\text{НДС} = C_{\Pi} * 12\% \quad (4.17)$$

$$\text{НДС} = 728567 * 0,12 = 87428 \text{ тенге}$$

В соответствии с формулой 4.16 цена реализации проекта составит

$$C_p = 728567 + 87428 = 815995 \text{ тенге}$$

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

#### **4.8 Вывод по технико-экономическому обоснованию**

Разработка программного обеспечения является трудным и кропотливым процессом, которая требует немалых затрат интеллектуального труда. Стоимость разработки включает в себя такие категории затрат, как фонд оплаты труда, отчисления на социальные нужды, амортизационные отчисления, затраты на электроэнергию.

Самую большую долю в общей себестоимости разработки программного обеспечения занимают затраты на оплату труда в размере 288 000,00 тенге, что составляет 47% от общей суммы разработки ПО.

На реализацию такого рода ПО, средней ценой услуг на рынке является цена, равной 1280000 тенге. Эта цена является итогом анализов фонд оплаты труда различных фирм, занимающихся разработкой и реализацией программ, с учетом такого же количества работников и дней на реализацию проекта, как и в данной дипломной работе. Анализируя полученную цену реализации проекта, равной 815995 тенге, можно сделать вывод о том, что данный проект является экономически выгодным.

## 5 БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ

### 5.1 Анализ условий труда

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

Данный проект ориентирован на улучшение работ медицинских учреждений, оказывающих первичную медико-санитарную помощь.

Цель работы - создание информационной системы, которая оптимизирует работу медицинских учреждений. То есть облегчит прием пациентов, сократив огромные очереди, уменьшит время, затрачиваемое на осмотр пациента.

Работа сотрудника напрямую связана с компьютером, а, следовательно, и с вредным воздействием группы факторов, что значительно понижает производительность труда. Компьютерная техника наносит большой вред здоровью сотрудника, поэтому работодателю необходимо обязательно учесть все опасные факторы на рабочем месте или при работе с техникой для безопасной работы своего сотрудника.

Работа сотрудников в помещении относится к лёгкой тяжести (1а), так как управление оборудованием осуществляется дистанционно с помощью компьютеров.

Для обеспечения соответствующих условий труда для сотрудников назначены нормы производственного микроклимата. В ходе работы с ПК необходимы соблюдаться следующие условия:

В холодный период:

- Оптимальная температура от 22 до 24 °С, допустимая – от 18 до 26 °С;
- Относительная влажность воздуха от 40 до 60%, допустимая 75%;

В теплый период:

- Оптимальная температура от 23 до 25 °С, допустимая от 20 до 30 °С;
- Относительная влажность воздуха от 40 до 60%, допустимая влажность 55%.

Расценим помещение, необходимое для разработки:

- рабочее помещение находится на 2 этаже;
- размеры рабочего помещения: длина 4 м, ширина 3 м, высота 3 м;
- остекление помещения – двойное (одно окно размером 2000 x 2000 мм);
- искусственное освещение – светильники: 2 светильника, в каждом по 2 люминесцентные лампы (ПВЛМ–1х40);
- по стандарту окраска кабинета и мебели должна создавать благоприятные условия для зрительного восприятия;
- для спасения от избыточной яркости на окнах установлены жалюзи;
- вид работы – разработка программного обеспечения для медицинских учреждений;

- количество рабочих мест – 2. Работа в дневную смену. Режим работы с 9:00 до 18:00, с перерывом на обед с 13:00 до 14:00.

План помещения представлен на рисунке 5.1.

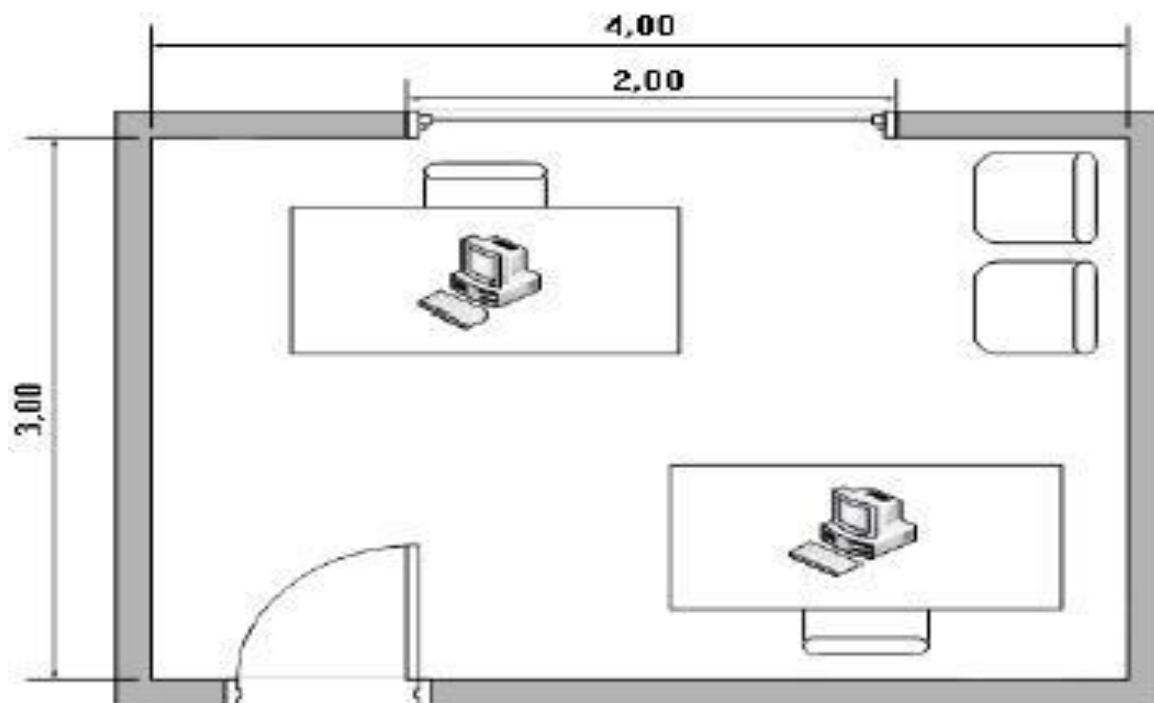


Рисунок 5.1 – План помещения

Рабочее место сотрудника снабжено следующим оборудованием:

- рабочий стол;
- стул (рабочий стул необходим быть подъемно-поворотным, регулируемым по высоте и углам наклона сиденья и спинки, а также расстоянию спинки от переднего края сиденья, при этом регулировка каждого параметра должна быть независимой, легко осуществляемой и иметь надежную фиксацию);
- персональный компьютер в стандартной комплектации (системный блок, монитор, клавиатура, компьютерная мышь);
- принтер.

Ниже изучена классическая офисная оргтехника и степень ее воздействия на сотрудника.

#### 1) Персональный компьютер.

Существует ряд вредоносных факторов, оказывающих влияние на работников, занятых на работе с видео-дисплейными терминалами (ВДТ) и персональными компьютерами (ПК):

- плохой микроклимат помещения: повышенная и пониженная температура воздуха; чрезмерная запыленность и загазованность воздуха; повышенная и пониженная влажность воздуха;

- мизерная освещенность рабочего места;
- шум, превосходящий допустимые нормы;
- завышенный уровень ионизирующего излучения;
- завышенный уровень электромагнитных полей;
- завышенный уровень статического электричества;
- опасность поражения электрическим током;
- блеклость экрана монитора;
- нарушение эргономических норм при работе с компьютером и др.

Низкочастотные электромагнитные поля и ионизирующее (рентгеновское) излучение мониторов на электронно-лучевых трубках (ЭЛТ) можно отнести к опасным излучениям компьютера.

Небольшая информация о радиационном излучении ЭЛТ-мониторов. Без сомнения, данный вид излучения присутствует в каждом приборе на электронно-лучевых трубках. Однако на расстоянии 50 см от экрана монитора уровень этого излучения не превышает уровня естественного фона. Кроме, мониторы на ЭЛТ едва ли не вытеснены жидкокристаллическими мониторами (ЖКИ), в которых отсутствует радиационное излучение.

При работе с ПК есть и более опасные вредные факторы. Все они не оказывают прямых опасностей, исходящими от этого вида техники, а являются эргономическими факторами, влияющими на здоровье человека.

Работа на персональных компьютерах относится к зрительно напряженным работам. Это значит, что в первую очередь при работе с компьютером страдают глаза человека. Конкретно этот вредный фактор присутствует в большинстве документов, регламентирующих труд работников с использованием ПЭВМ.

Существуют три группы работ с видео-дисплейными терминалами и ПЭВМ, к тому же три категории сложности и напряженности работы. К первой группе работ (группа А) относятся работы по считыванию информации с дисплея, ко второй (группа Б) относятся работы по вводу информации, к третьей (группа В) относятся интерактивная работа с компьютером. Бесспорно, труд офисного работника в основном относится к группе В, потому что требует интерактивной работы с различным программным обеспечением.

На зрительную усталость безмерно сильно влияет также уровень освещенности рабочего места. Заметно это, особенно, при одновременной работы с электронными и бумажными документами. Уровень освещенности рабочего места при работе за компьютером должен составлять 300-500 лк. При этом монитор и источники света должны быть расположены таким образом, чтобы не создавать бликов на поверхности экрана.

Последующей опасностью, подстерегающей работников в офисе,

является неподвижность позы при работе за компьютером. Неподвижная напряженная поза при длительной работе на компьютере может привести к воспалению мышц, связок и сухожилий спины и ног, заболеваниям позвоночника и суставов (остеохондроз и пр.), а постоянное напряжение рук - к повреждениям запястья и сухожилий (так называемый синдром луче-запястного сустава или туннельный синдром). Эти заболевания вызываются так называемыми травмами повторяющихся нагрузок, которые являются постепенно накапливающимися недомоганиями, обусловленные продолжительными повторяющимися воздействиями и перетекающие в болезни нервов, мышц и сухожилий. Влияние данного вредного фактора понижается при правильной организации рабочего места – оптимально подобранной мебели, правильном размещении элементов компьютера.

## 2) Принтер.

Если работа с персональным компьютером во многом регламентирована законодательно, то этого уже не скажешь об остальном оборудовании, используемом в офисе. В отношении принтера, то ясно: его использование не обязывает постоянного присутствия работника, напряжения внимания и т.п. Вся работа с принтером сводится к его включению и выключению, добавлению бумаги и изыманию отпечатков. Даже замену картриджа во многих случаях выполняет не пользователь, а обслуживающий персонал.

Впрочем, любой принтер является сложным электроприбором, оттого при работе с ним необходимо выполнять стандартные требования пожарной и электробезопасности. Также в любом принтере имеются движущиеся части, а некоторые элементы могут нагреваться в процессе работы до высокой температуры (в матричных и струйных принтерах это печатные головки, в лазерных принтерах – специальные нагревательные валы).

Еще одним вредным фактором при работе с принтерами является шум. Обладателями наивысшего уровня шума во время работы являются матричные принтеры, однако в паспортах этих приборов уровень шума, как правило, не указан, а фактический уровень может быть измерен ежели при проведении аттестации рабочего места. Это единственный тип принтеров, уровень шума которых может быть сопоставим с максимально допустимым на рабочих местах, оборудованных ПЭВМ. Все остальные принтеры работают заведомо тише.

Шум портит условия труда, оказывая вредное действие на организм человека. Работающие в условиях длительного шумового воздействия испытывают раздражительность, головные боли, головокружение, снижение памяти, повышенную утомляемость, понижение аппетита, боли в ушах и т. д. Такие нарушения в работе ряда органов и систем организма человека могут вызвать негативные изменения в эмоциональном состоянии человека вплоть до стрессовых. Под воздействием шума снижается концентрация внимания, нарушаются физиологические функции, появляется усталость в связи с повышенными энергетическими затратами и нервно-психическим напряжением,

ухудшается речевая коммутация. Все эти факторы понижают работоспособность человека и его производительность, качество и безопасность труда. Воздействие интенсивного шума на слух человека на длительной основе [выше 80 дБ(А)] приводит к его частичной или полной потере.

В табл. 1.1 указаны предельные уровни звука в зависимости от категории тяжести и напряженности труда, являющиеся безопасными в отношении сохранения здоровья и работоспособности.

Таблица 1.1 Предельные уровни звука, дБ, на рабочих местах.

Категория напряженности труда	Категория тяжести труда			
	I. Легкая	II. Средняя	III. Тяжелая	IV. Очень тяжелая
I. Мало напряженный	80	80	75	75
II. Умеренно напряженный	70	70	65	65
III. Напряженный	60	60	-	-
IV. Очень напряженный	50	50	-	-

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

#### Окраска и коэффициенты отражения

Для хорошего настроения и зрительного восприятия, окраска помещения и мебели должна способствовать созданию благоприятных условий.

Значительно ухудшают точность знаков и влекут за собой помехи физиологического характера, которые могут выразиться в значительном напряжении, особенно при продолжительной работе источники света, например, светильники и окна, которые дают отражение от поверхности экрана. Чтобы защититься от избыточной яркости окон можно применить шторы и экраны.

Рекомендуется окраску стен и пола выбирать в зависимости от ориентации окон.

Окна, ориентированные на юг:

- стены зеленовато-голубого или светло-голубого цвета; пол - зеленый;

Окна, ориентированные на север:

- стены светло-оранжевого или оранжево-желтого цвета; пол - красновато-оранжевый;

Окна, ориентированные на восток:

- стены желто-зеленого цвета;
  - пол зеленый или красновато-оранжевый;
- Окна, ориентированные на запад:
- стены желто-зеленого или голубовато-зеленого цвета;
  - пол зеленый или красновато-оранжевый.

В компьютерных помещениях нужно обеспечить нижеследующие величины коэффициента отражения: для потолка: 60,70%, для стен: 40,50%, для пола: около 30%. Для других поверхностей и рабочей мебели: 30,40%.

### **Эргономические требования к рабочему месту**

К числу важных проблем эргономического проектирования в области вычислительной техники, проектирование рабочих мест, снабженных видеотерминалами, относится к числу важных проблем. Рабочее место сотрудника и взаимное расположение элементов должно быть соответствующим антропометрическим, физическим и психологическим запросам. Огромное значение имеет также характер работы. В основном, в ходе организации рабочего места программиста нужно соблюсти следующие основные условия: оптимальное размещение оборудования, входящего в состав рабочего места и достаточное рабочее пространство, позволяющее осуществлять все необходимые движения и перемещения.

В основном, эргономическими аспектами проектирования рабочих мест являются: высота рабочей поверхности, размеры свободного пространства для ног, расстояние от глаз сотрудника до экрана, документа, клавиатуры, характеристики рабочего кресла, требования к поверхности рабочего стола, регулируемость элементов рабочего места. У сотрудника вызывает минимальное утомление рабочая поза сидя. Рациональная планировка рабочего места предусматривает четкий порядок и постоянство размещения предметов, средств труда и документации. То, что требуется для выполнения работ чаще, расположено в зоне легкой досягаемости рабочего пространства.

### **5.2 Расчет системы кондиционирования офиса**

Ниже обусловлены расчеты системы кондиционирования в помещении. Кондиционирование предоставляет наилучший микроклимат в помещении и условия работы точной и чувствительной аппаратуры.

Для организации комфортных условий труда в помещении, следует определить необходимое количество кондиционеров. За счёт тепловыделений производственного оборудования в помещении могут присутствовать значительные избытки тепла (разность между тепловыделениями в помещении и теплоотдачей через стены, окна, двери и т.д.), удаление которых, прежде всего, должна обеспечить система вентиляции.

Избыточное тепло рассчитывается по формуле:

$$Q_{\text{ИЗБ}} = (Q_{\text{ОБ}} + Q_{\text{ОСВ}} + Q_{\text{Л}} + Q_{\text{Р}}) - Q_{\text{ОТД}} \quad (5.1)$$

где,  $Q_{\text{ОБ}}$ ,  $Q_{\text{ОСВ}}$ ,  $Q_{\text{Л}}$  – тепло, выделяемые производственным оборудованием, системой искусственного освещения помещения и работающим персоналом (людьми) соответственно, ккал/ч;

$Q_{\text{Р}}$  – тепло от солнца, освещающее помещение (солнечная радиация), ккал/ч;

$Q_{\text{ОТД}}$  – теплоотдача естественным путём, ккал/ч.

Тепло, выделяемое производственным оборудованием рассчитывается по формуле:

$$Q_{\text{ОБ}} = 860 \times P_{\text{ОБ}} \times \eta \quad (5.2)$$

где, 860 – тепловой эквивалент, равный 1 кВт/ч;

$P_{\text{ОБ}}$  – мощность, потребляемая оборудованием, кВт/ч;

$\eta$  – коэффициент перехода тепла в помещение. Значение  $\eta = 0,75$  – норма потерь потребляемой мощности на тепловыделения компьютерного оборудования.

Для 2 компьютеров имеем:

$$Q_{\text{ОБ}} = 860 \times (2 \times 0,25) \times 0,75 = 322,5 \text{ ккал/ч}$$

Тепло, выделяемое осветительными установками, рассчитывается по формуле:

$$Q_{\text{ОСВ}} = 860 \times N \times \eta \quad (5.3)$$

где,  $N$  – расходуемая мощность светильников, кВт;

$\eta = 0,45$  – норма потерь потребляемой мощности на тепловыделения люминесцентных ламп.

$$Q_{\text{ОСВ}} = 860 \times 0,45 \times 0,54 = 209 \text{ ккал/ч}$$

Тепло, выделяемое людьми, рассчитывается по формуле:

$$Q_{\text{Л}} = K_{\text{Л}} \times q \quad (5.4)$$

где,  $K_{\text{Л}}$  – количество работающих людей;

$q$  – тепловыделения одного человека, ккал/ч.

$q = 60$ .

$$Q_{\text{Л}} = 2 \times 60 = 120 \text{ ккал/ч}$$



Тепло, вносимое солнечной радиацией, рассчитывается по формуле:

$$Q_p = m \times F \times q_{\text{ост}} \quad (5.5)$$

где,  $m$  – количество окон в помещении;

$F$  – площадь одного окна,  $\text{м}^2$ ;

$q_{\text{ост}}$  – солнечная радиация через остеклённую поверхность, т.е. количество тепла, вносимое за один час через остеклённую поверхность площадью в  $1 \text{ м}^2$ .

Для окна с двойным остеклением с деревянными переплетами  $q_{\text{ост}} = 107$ . Количество окон равно 1. Площадь окна равна  $1 \times 2 = 2 \text{ м}^2$ .

$$Q_p = 1 \times 2 \times 107 = 214 \text{ ккал/ч}$$

Для тёплого периода года при расчётах можно принять  $Q_{\text{отд}} = 0$ .

$$Q_{\text{изб}} = 322,5 + 209 + 120 + 214 = 865,5 \text{ ккал/ч}$$

Необходимое количество воздуха, при наличии теплоизбытков, которое нужно удалить из помещения, рассчитывается по формуле:

$$L_b = \frac{Q_{\text{изб}}}{c_b \times \Delta t \times \gamma_b} \quad (5.6)$$

где,  $Q_{\text{изб}}$  – избыточное тепло, ккал/ч;

$c_b$  – теплоёмкость воздуха ( $0,22 \text{ ккал/кг}^\circ\text{C}$ );

$$\Delta t = t_{\text{вых}} - t_{\text{вх}};$$

$$\Delta t = 8^\circ\text{C};$$

$t_{\text{вых}}$  – температура воздуха, выходящего из помещения,  $^\circ\text{C}$ ;

$t_{\text{вх}}$  – температура воздуха, поступающего в помещение,  $^\circ\text{C}$ ;

$\gamma_b = 1,204 \text{ кг/м}^3$  – удельная масса приточного воздуха.

$$L_b = \frac{865,5}{0,22 \times 8 \times 1,204} = 408,3 \text{ м}^3/\text{ч}$$

Отношение количества воздуха, поступающего в помещение за один час, к объему помещения называется кратностью воздухообмена:

$$K = \frac{L}{V_n} \quad (5.7)$$

где,  $V_n$  – объем помещения,  $\text{м}^3$ ;

$$V_n = 4 \times 3 \times 3 = 36 \text{ м}^3.$$

$$K = \frac{408,3}{36} \approx 11,3$$

Существующий оконный кондиционер имеет расход воздуха  $670 \text{ м}^3/\text{ч}$ . Определим требуемое количество таких кондиционеров:

$$N = \frac{408,3}{670} \approx 1 \text{ кондиционер}$$

Тем самым соответствует действительности и является достаточным для обеспечения комфортного микроклимата.

### 5.3 Пожарная безопасность

Само здание по степени опасности развития пожара, от функционального назначения и пожарной нагрузки горючих материалов, относится к 1 – ой группе категории D.

Причинами возникновения пожара могут быть:

- возгорание элементов аппаратуры;
- возгорание отделочных материалов от неисправных выключателей, розеток;
- несоблюдение режимов эксплуатации оборудования, неправильные манипуляции персонала.

При появлении пожара может пострадать не только помещение, но и дорогостоящие устройства, являющиеся опасными для жизни человека. Поэтому, есть необходимость того, чтобы были приняты меры по раннему выявлению и устранению пожаров. Источниками зажигания могут оказаться персональные компьютеры, приборы, устройства электропитания, кондиционеры воздуха, где в результате разного рода нарушений, образуются перегретые элементы и прочее.

Согласно требованиям правил пожарной безопасности, помещение оборудовано углекислотными огнетушителями ОУ – 5 с учётом – один огнетушитель на 100 м<sup>2</sup>. Общая площадь помещения составляет 12 м<sup>2</sup> таким образом устанавливаются 1 огнетушитель. В качестве огнетушащего вещества применяется комбинированный углекислотно – хладоновый состав. Расчётная масса комбинированного углекислотно – хладонового состава  $m_d$ , кг, для объёмного пожаротушения определяется по формуле

$$m_d = k \times g_n \times V \quad (5.8)$$

где,  $k = 1,4$  – коэффициент компенсации не учитываемых потерь углекислотно–хладонового состава;

$g_n = 0,06$  – нормативная массовая концентрация углекислотно–хладонового состава;

$V$  – объём помещения.

Объём помещения определяется по формуле

$$V = A \times B \times H \quad (5.9)$$

где  $A = 4$  м – длина помещения;

$B = 3$  м – ширина помещения;

$H = 3$  м – высота помещения.

Тогда

$$V = 4 \times 3 \times 3 = 36 \text{ м}^3.$$

Следовательно

$$m_d = 1,4 \times 0,06 \times 36 = 3 \text{ кг}.$$

Расчётное число баллонов  $\xi$  определяется из расчета вместимости в 20 - литровый баллон 14 кг углекислотно–хладонового состава. Внутренний диаметр магистрального трубопровода  $d_i$ , мм равен

$$d_i = 14 \times \sqrt{2} = 20 \text{ мм}$$

Эквивалентная длина магистрального трубопровода  $l_2$ , м определяется по формуле

$$l_2 = k_1 \times l \quad (6.0)$$

где,  $k_1 = 1,3$  – коэффициент увеличения длины трубопровода для компенсации не учитывающих местных потерь;

$l = 3$  м – длина трубопровода по проекту тогда

$$l_2 = 1,3 \times 3 = 3,9 \text{ м.}$$

Расход углекислотно–хладонового состава  $Q$ , кг/с, в зависимости от эквивалентной длины и диаметра трубопровода равна 1,6 кг/с.

Расчётное время подачи углекислотно–хладонового состава  $t$ , мин, определяется по формуле

$$t = \frac{md}{60 \times Q} \quad (6.1)$$

Таким образом,

$$t = \frac{3}{60 \times 1,6} = 0,03$$

Масса основного запаса углекислотно–хладонового состава  $m$ , кг, определяется по формуле

$$m = 1,1 \times m_d \times \left(1 + \frac{k_2}{k}\right) \quad (6.2)$$

где,  $k_2 = 0,4$  – коэффициент учитывающий остаток углекислотно–хладонового состава в баллонах и трубопроводах

$$m = 1,1 \times 3 \times \left(1 + \frac{0,4}{1,4}\right) = 4 \text{ кг.}$$

Из полученных результатов следует, что для обеспечения нормального функционирования системы автоматического пожаротушения потребуется 1 баллон углекислотно–хладонового состава вместимостью 20 литров, с массой смеси 4 кг. Автоматические установки газового пожаротушения имеют устройства для автоматического пуска в соответствии с ГОСТ 12,4.009–83.

Нахождение огнетушителя представлено на рисунке 5.2.

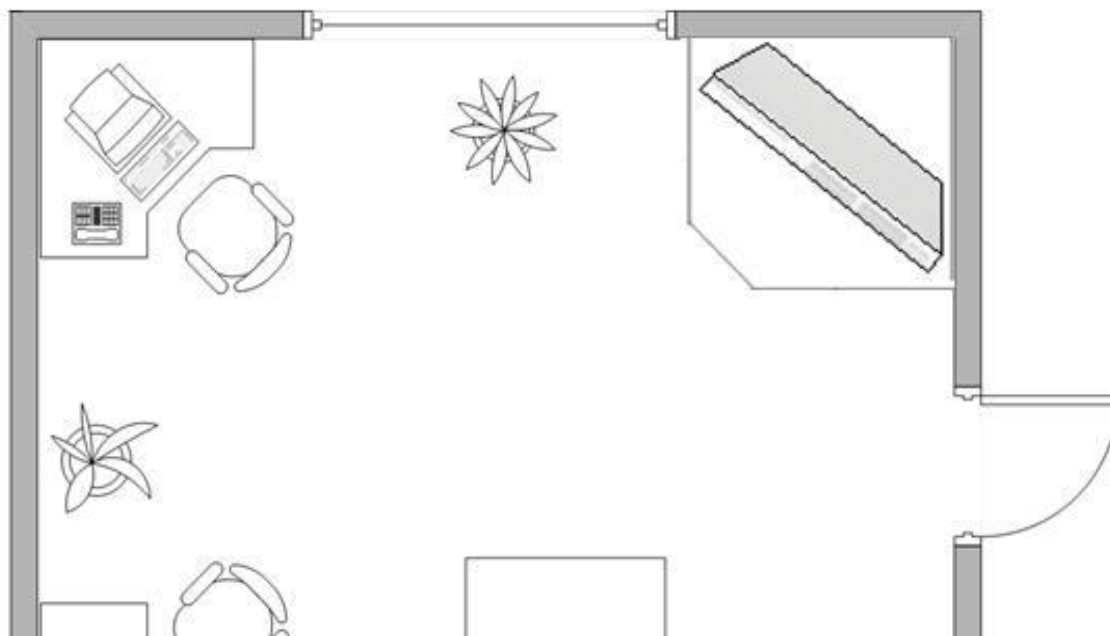


Рисунок 5.2 – Размещение огнетушителя в помещении

#### **5.4 Вывод по части безопасности жизнедеятельности**

В конечном итоге проделанных работ просчитаны необходимые меры безопасности и условия труда разработчика-программиста информационного оборудования, которые соответствуют стандартам СНиП, и ГОСТ.

В помещениях потенциальной угрозой возникновения пожара является электротехническое оборудование. Из полученных расчётов выводом является, что для предоставления нормального функционирования системы автоматического пожаротушения потребуется 1 баллон углекислотно-хладонового состава вместимостью 20 литров, с массой смеси 4 кг.

Опираясь на расчеты, в помещение с оборудованием, для соблюдения требуемых параметров микроклимата необходимо установить один кондиционер с производительностью не менее 670 м<sup>3</sup>/ч.

## **Заключение**

В данной дипломной работе было разработано программное обеспечение для медицинских учреждений с помощью технологии С#. В ходе всей работы были соблюдены и выполнены поставленные требования к программному продукту. В качестве используемой технологии были взяты язык программирования С# в среде разработки Visual Studio с использованием сопутствующих технологий, таких как база данных MS SQL Server и ASP.NET.

Была изучена модель предметной области, проведен опрос среди поликлиник, произведен статистический анализ данных, выявлены причин проблемы.

Следующим шагом стала разработка схемы баз данных, создание таблиц, столбцов и полей, ключей и связей. Заполнение базы данных. После создания соединили базу данных с графической оболочкой.

Далее проработали структуру алгоритма работы системы, создали формы, пользователей, добавили нужную библиотеку для создания базы знаний. Детально спроектировали пользовательский интерфейс для системы. Каждый пользователь может авторизоваться со своим личным паролем. Создали базу знаний для генерации информации, которая выдает перечень обследований для пациента. После чего, специалист может направить пациента в лабораторию. В свою очередь лаборатория принимает запросы на анализы, после заносятся готовые анализы в базу данных. В самом конце врач может выдать готовую справку пациенту, которая была сделана в виде шаблона.

На последнем этапе было произведено тестирование программного обеспечения. На данный момент система является наиболее актуальной и рекомендуется для установки в медицинских учреждениях, оказывающих амбулаторный прием.

### Список используемой литературы

1. Бабэ, Бруно Просто и ясно о Borland C++; М.: БИНОМ, 1995. - 400 с.
2. Джейсон, Прайс; Майк, Гандэрлой Visual C# .NET. Полное руководство; КОРОНА принт, 2004. - 960 с.
3. Робинсон, С.; Корнес, О.; Глинн, Д. и др. C# для профессионалов; М.: Лори, 2005. - 396 с.
4. Алпатов А.П., Прокопчук Ю.А., Костра В.В. Госпитальные информационные системы: архитектура, модели, решения. - Днепропетровск: УГХТУ, 2005. - 257 с.
5. Воронцов И.М., Шаповалов В.В., Шерстюк Ю.М. ЗДОРОВЬЕ. Создание и применение автоматизированных систем для мониторинга и скринирующей диагностики нарушений здоровья. - Санкт-Петербург: 2006.
6. Назаренко Г. И., Осипов Г. С. Медицинские информационные системы и искусственный интеллект. Вып. 3: Науч. пособ. М.: Медицина XXI, 2003. - 320 с.
7. Гаврилова – Базы знаний интеллектуальных систем: Пособие. 2003. – С 234.
8. Ахметова М. -Системы искусственного интеллекта: Конспект лекций. 2013. – 46 с.
9. Тихонов Э.Е. Прогнозирование в условиях рынка. Невинномысск, 2006. - 221 с.
10. Хорошевский В. Базы знаний интеллектуальных систем. Учебник. — СПб.: Питер, 2000, - 225 с.
11. Джонс М. Т. Программирование искусственного интеллекта в приложениях М. Тим Джонс; Пер. с англ. Осипов А. И. — М.: ДМК Пресс, 2006. - 312 с.

12. Сатимова Е.Г. Проектирование баз данных. Методические указания к выполнению лабораторных работ (для студентов всех специальностей) – Алматы: АИЭС, 2009. - 37с.

13. Сербин В.В. ИСРП. Методические указания к выполнению лабораторных работ - Алматы: АУЭС, 2013. - 21с.

14. Методические указания к выполнению экономической части дипломных работ для студентов специальности 5В070400 – Вычислительная техника и программное обеспечение. З.Д. Еркешева, Г.Ш. Боканова Алматы: АУЭС, 2013 – 40 с.

15. ГОСТ 12.1.005-88 ОБЩИЕ САНИТАРНО-ГИГИЕНИЧЕСКИЕ ТРЕБОВАНИЯ К ВОЗДУХУ РАБОЧЕЙ ЗОНЫ

16. ГОСТ 30494-96 ПАРАМЕТРЫ МИКРОКЛИМАТА В ПОМЕЩЕНИЯХ

17. habrahabr.ru – Написание парсера с нуля, 21.11.13

18. kazpravda.kz – Минздрав проанализировал причины очередей в поликлиниках, 10.07.15

19. uvision.kz – Простой граббинг

## Приложение А

### Листинг программной части

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 label11_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 label11_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)
{
}
}
}

```

1. using System.Text;
- 2.
- 3.
4. **namespace** ConsoleApplication1
5. {
6. using HtmlAgilityPack;
7. using System;
8. using WatiN.Core;
- 9.
10. **class** Program
11. {
12. [STAThread]

```

13. static void Main()
14. {
15. Console.OutputEncoding = Encoding.UTF8;
16.
17. using (var browser = new FireFox("http://www.yvision.kz"))
18. {
19. Settings.WaitForCompleteTimeOut = 999999999;
20. var doc = new HtmlDocument();
21. doc.LoadHtml(browser.Body.OuterHtml);
22. var dataBlock = doc.DocumentNode.SelectSingleNode("//div[@class=\"mainContent
    main_page\"]"); /*получаем в переменную все блоки из дива .mainContent.mainpage*/
23.
24. foreach (var item in dataBlock.SelectNodes("//div[@class=\"half-1\"]")) /*в цикле
    проходим по каждому диву с классом .half-1*/
25. {
26. Console.Write( item.InnerHtml + "\n"); /*показываем в консоли содержимое дива*/
27. }
28. }
29. Console.WriteLine("Done!");
30. Console.ReadKey();
31. }
32. }
33. }

```

```

1. using System.Text;
2.
3.
4. namespace ConsoleApplication1
5. {
6. using HtmlAgilityPack;
7. using System;
8. using WatiN.Core;
9.
10. class Program
11. {
12. [STAThread]
13. static void Main()
14. {
15. Console.OutputEncoding = Encoding.UTF8;
16.
17. using (var browser = new FireFox("http://www.yvision.kz"))
18. {
19. Settings.WaitForCompleteTimeOut = 999999999;
20. var doc = new HtmlDocument();
21. doc.LoadHtml(browser.Body.OuterHtml);
22. var dataBlock = doc.DocumentNode.SelectSingleNode("//div[@class=\"mainContent
    main_page\"]");
23. int cntItem = 0; /*добавим счетчик, мы ведь будем выводить только 4 записи*/
24. foreach (var item in dataBlock.SelectNodes("//div[@class=\"half-1\"]"))

```

```
25. {
26. cntItem++; /*увеличиваем счетчик на один*/
27. Console.WriteLine("Link item:
    " +item.SelectSingleNode("//div[@class=\"info\"]/div[@class=\"photo\"]/div[@class=\"
    hold\"]/a").Attributes["href"].Value+ "\n");
28. Console.WriteLine("Images patch:
    " +item.SelectSingleNode("//div[@class=\"info\"]/div[@class=\"photo\"]/div[@class=\"
    hold\"]/a/img").Attributes["src"].Value+ "\n");
29. Console.WriteLine("\n*****yvision.kz*****\n\n"); /*визуально разграничим результаты
    текстом*/
30. if(cntItem == 4) {break;} /*если выбрали 4 записи то остановить цикл*/
31. }
32. }
33. Console.WriteLine("Done!");
34. Console.ReadKey();
35. }
36. }
37. }
```