

Министерство образования и науки Российской Федерации
Филиал Федерального государственного автономного образовательного учреждения
высшего образования
«Южно-Уральский государственный университет»
(национальный исследовательский университет)
в г. Нижневартовске

Кафедра «Информатика»

РАБОТА ПРОВЕРЕНА

ДОПУСТИТЬ К ЗАЩИТЕ

РЕЦЕНЗЕНТ

Исп. Дир. ООО НТП «НГТ»

И.о. зав. кафедрой «Информатика»

к.т.н., доцент

/Ф.Х.Азизов

/ Н.И.Юмагулов

«__» _____ 2017 г.

«__» _____ 2017 г.

Разработка информационной системы для сбора и обработки данных опроса и анкетирования

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
ЮУрГУ-09.03.04.2017.107.ПЗ ВКР

Консультанты

Экономическая часть

к.э.н., доцент

/А.В.Прокопьев

«__» _____ 2017г.

Руководитель работы

к.т.н., доцент

/ И.Л.Кафтанников /

«__» _____ 2017 г.

Безопасность жизнедеятельности

к.т.н., доцент

/ Н.И.Юмагулов

«__» _____ 2017г.

Автор работы

обучающийся группы НвФл-422

/ Э.И. Трофимов /

«__» _____ 2017г.

Нормоконтролер

старший преподаватель

/Л.Н. Буйлушкина/

«__» _____ 2017г.

Нижневартовск 2017

АННОТАЦИЯ

Трофимов Э.И. Разработка информационной системы для сбора и обработки данных опроса анкетирования – Нижневартовск: филиал ЮУрГУ, Информатика: 2017, 83 с., 12 ил., 11 табл., библиогр. список – 17 наим., 2 прил.

Разработана информационная система для сбора и анализа данных анкетирования.

Цель выпускной квалификационной работы заключается в создании информационной системы для сбора и обработки данных анкетирования. Система предназначена сократить время и ресурсы на сбор и обработку данных анкетирования, уменьшить объем бумажной документации, а также избежать ошибок при обработке данных, вызванных человеческим фактором.

Задачи:

1. Создание базы данных для хранения данных анкетирования;
2. Разработка формы анкеты для студентов;
3. Разработка приложения для отображения результатов анкетирования;
4. Внедрение системы;
5. Разработка руководства пользователя.

Рассмотрены методы разработки программных продуктов. Большой акцент ставился на самостоятельном изучении и применении при разработке технологий доступа к базам данных. Разработано приложение, осуществляющее процесс сбора и обработки данных.

Изм	Лист	№ докум.	Подпись	Дата				
					09.03.04.2017.107.ПЗ			
Разработал		Трофимов Э.И.			Разработка информационной системы для сбора и обработки данных опроса и анкетирования	Лит.	Лист	Листов
Проверил		Юмагулов Н.И.				17	6	79
Рецензент		Азизов Ф.Х.				Филиал ФГАОУ ВО «ЮУрГУ (НИУ)» в г. Нижневартовске кафедра «Информатика»		
Н.контр.		Буйлушкина Л.Н.						
Утвердил		Юмагулов Н.И.						

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	7
1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПОСТАНОВКА ЗАДАЧИ	9
1.1 Анализ проблемы.....	9
1.2 Понятие и характеристики информационной системы	11
1.3 Требования к разрабатываемой системе	13
1.4 Выбор средств разработки	13
1.5 Выбор СУБД.....	20
2 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА БД И ПРИЛОЖЕНИЯ ДЛЯ АНКЕТИРОВАНИЯ	24
2.1 Разработка БД.....	24
2.2 Разработка приложения.....	30
3 ОРГАНИЗАЦИОННО-ЭКОНОМИЧЕСКИЙ РАЗДЕЛ	39
3.1 Расходы на приобретение, содержание и эксплуатацию программного и аппаратного обеспечения	40
3.2 Расчет экономического эффекта	45
4 БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ	51
4.1 Требования к помещениям при работе за компьютером.....	51
4.2 Требования к освещению помещений и рабочих мест	51
4.3 Требования к микроклимату	52
4.4 Требования к шуму и вибрации в помещениях	53
4.5 Требования к рабочему месту.....	55
4.6 Режим труда и отдыха при работе за компьютером	57
ЗАКЛЮЧЕНИЕ	62
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	63
ПРИЛОЖЕНИЯ	
ПРИЛОЖЕНИЕ А. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ	65
ПРИЛОЖЕНИЕ Б. КОМПАКТ-ДИСК.....	84

ВВЕДЕНИЕ

Одной из проблем эффективной организации работы в высшем учебном заведении является задача автоматизации процесса проведения анкетирования.

Вне зависимости от объекта автоматизации, будь то преподавательский состав или администрация университета, в образовательном учреждении такие системы внедряют, преследуя конечную цель – повышение качества образования.

ВУЗ, как и любое предприятие, непременно проходит процесс автоматизации и, несмотря на то, что понятие образовательной деятельности едино для всех образовательных учреждений, в каждом ВУЗе этот процесс проходит по-разному. Значительное влияние на процессы автоматизации оказывает как наличие денежных средств, так и готовность использования предлагаемых рынком информационных услуг программных продуктов.

С целью снижения материальных и трудовых затрат в филиале ФГАОУ ВО «ЮУрГУ (НИУ)» в г. Нижневартовске (далее – филиал ЮУрГУ) было разработано решение, автоматизирующее процесс проведения анкетирования среди обучающихся университета и выполняющего обработку полученных данных.

Цель выпускной квалификационной работы заключается в создании информационной системы для сбора и обработки данных анкетирования.

В соответствии с этой целью были определены следующие задачи:

- Создание базы данных для хранения данных анкетирования;
- Разработка формы анкеты для студентов;
- Разработка приложения для отображения результатов анкетирования;
- Внедрение системы.

Объектом является процесс анкетирования студентов.

Предметом является разработка информационной системы для сбора и обработки данных анкетирования.

Методологической основой исследования являются анализ, сравнение, абстрагирование, классификация, а также использование дедуктивного и индуктивного методов научного познания.

Эмпирическую базу исследования составили:

- собственный опыт прохождения анкетирования;
- интервьюирование заместителя директора по учебной практике.

Общие требования, предъявляемые к автоматизированной системе, следующие:

- быстрота обработки информации за счет автоматизации возможных операций пользователя системы;
- формирование отчетных форм;
- расширяемость системы (возможность её доработки в случае повышения требований к автоматизированной системе);
- удобный пользовательский интерфейс.

Практическая значимость данной работы состоит в снижении материальных и трудовых затрат при проведении анкетирования среди обучающихся филиала ЮУрГУ.

В первом разделе выпускной квалификационной работы рассматривается анализ проблем, связанных с автоматизацией процесса анкетирования, рассматриваются основные подходы к разработке программного обеспечения (далее – ПО), выбор среды разработки и базы данных (далее – БД).

Во втором разделе формулируются требования к разрабатываемой системе, реализация БД и самой программы.

В третьем разделе описывается охрана труда и безопасность жизнедеятельности для пользователей ПО.

В четвертом разделе рассчитываются затраты на разработку данного ПО, и последующая выгода от его использования.

1 АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ И ПОСТАНОВКА ЗАДАЧИ

1.1 Анализ проблемы

Система оценки персонала – система способов измерения стоимости основного человеческого ресурса. Экономически она характеризует эффективность ресурса с точки зрения целевой функции (критерия оптимальности). Благодаря ее использованию достигается наилучшее использование имеющихся ресурсов.

Система оценки персонала включает в себя следующие виды оценки:

Аттестация объекта (работника) – процедура определения ценности (цены) направленная на подведение итогов деятельности объекта объединенная с анализом возможности достижения им плановых показателей в будущем.

Оценка объекта (работника) – проведение контрольных мероприятий направленных на контроль деятельности объекта, которые заключаются в том, чтобы убедиться, что были должным образом, выбраны и применены методы управленческого воздействия на объект.

Человеческие ресурсы предприятия – это экономически активный, трудоспособный персонал организации, который позволяет работодателю в настоящее время извлекать максимум полезности из имеющихся у него знаний, умений и навыков.

Если раньше в основном работодатели пытались оценить одну из функций (лояльность, профессионализм и т.д.), то сейчас все больше и больше руководителей интересует, как оценить совокупный человеческий ресурс предприятия. При этом оценка человеческих ресурсов иногда заменяется оценкой кадрового потенциала (как составляющей стоимости предприятия). Это не совсем верно, так как для оценки стоимости компании важно не то, какую прибыль она может получить в будущем, а то, что она имеет сегодня и сейчас. Именно поэтому

оценка человеческого ресурса так важна для оценки стоимости активов предприятия.

Метод анкетирования – один из наиболее распространенных методов оценки. Оценочная анкета состоит из определенного набора вопросов и описаний. Оценивающий анализирует наличие или отсутствие указанных черт у оцениваемого, отмечая подходящий вариант.

В филиале ЮУрГУ ежегодно проходит оценка качества работы преподавательского состава. Обучающиеся филиала оценивают работу преподавателей по определенным критериям, и на основании этого составляется итоговая оценка.

Таким образом, анкетирование «Преподаватель глазами студента» является приемлемым методом объективного оценивания работы преподавателя. Однако сам процесс анкетирования нагружает преподавателей и методистов лишней работой, обработка результатов проводится вручную, что может повлечь за собой возможные ошибки при обработке данных анкетирования.

Автоматизации процесса анкетирования может сократить приведенные недостатки этого метода оценивания персонала организации. К сожалению, на рынке нет такой автоматизированной системы, которая бы удовлетворила потребности филиала ЮУрГУ.

В связи с расширением работ по совершенствованию системы управления высшей школой путем создания и внедрения в ВУЗах различных автоматизированных систем возникла потребность в разработке унифицированной информационной системы автоматического анкетирования на вычислительной технике. Для этого необходимо четко формализовать требования к анкетированию и разработать соответствующее алгоритмическое обеспечение.

1.2 Понятие и характеристики информационной системы

Информационные технологии – весьма широкое определение, под которое попадает ряд отдельных технических средств и приемов работы с информацией. Но, как правило, в процессе работы с информацией люди имеют дело со вполне определенной связанной последовательностью взаимодействий с различными средствами. В зависимости от шкалы времени (от оперативных до долгосрочно-стратегических задач) и масштаба действий (от одного рабочего места до целой компании) могут обнаруживаться разные связи и последовательности, и для управления ими нужны разные методы. Методы варьируются также по степени алгоритмизации и рационализации. Наиболее рационально алгоритмизированная совокупность методов и средств работы с информацией - информационная система.

Информационная система (далее – ИС) – взаимосвязанная совокупность средств, методов и персонала, используемых для хранения, обработки и выдачи информации. Информационные системы бывают разного назначения и масштаба. Также информационные системы отличаются по степени охвата сфер деятельности предприятия (учитывают ли они только бухгалтерию или также и склад, финансы, производство и т.д.). Однако все информационные системы обладают рядом свойств, которые являются для них общими:

– ИС предназначены для сбора, хранения и обработки информации. Таким образом, в основе любой информационной системы лежат средства хранения и доступа к данным.

– ИС предназначены для конечного пользователя, не являющегося специалистом в области вычислительной техники. Из этого следует, что ИС должны включать в себя клиентские приложения, обеспечивающие интуитивно понятный интерфейс.

ИС бывают фактографическими и документальными. К первым относятся ИС, предназначенные для поиска однозначного ответа на запрос и для

однозначного решения поставленных задач. Условно фактографические ИС делят в свою очередь на информационно-справочные системы, информационно-поисковые системы и системы оперативной обработки данных. Системы оперативной обработки данных решают такие задачи, как управление производством, бухгалтерский учет и т.п. По масштабу ИС делятся на:

- настольные;
- сетевые;
- ИС масштаба предприятия.

Документальные информационные системы предназначены для решения задач, не предусматривающих однозначного ответа на вопрос. Некоторые системы представляют собой смешанный тип фактографической и документальной ИС.

В большинстве случаев, когда требуется комплексная система, для хранения, обработки и поиска информации, процедура внедрения оказывается очень длительной и трудоемкой. Но несмотря на то, что процесс освоения предприятием такого нововведения, как ИС весьма сложен, в случае успешного завершения процесса затраченные ресурсы затем всегда окупаются за счет того, что ИС:

- автоматизирует применение математических методов к решению управленческих задач;
- частично освобождает сотрудников от рутинного труда;
- минимизирует вероятность появления ошибки в ходе передачи либо обработки информации;
- снижает объем бумажных документов;
- совершенствует документооборот;
- снижает затраты на производство товаров и услуг.

1.3 Требования к разрабатываемой системе

Одной из целей, преследуемых при проведении автоматизации ВУЗа, является создание единой ИС. Наличие единого центра хранения информации с целью минимизировать функции отдельных пользователей – не менее важная задача при создании автоматизированных систем.

При этом при разработке системы необходимо учитывать оборудование и программное обеспечение, которое уже имеется в филиале ЮУрГУ, для того, чтобы минимизировать расходы на приобретение и установку дополнительного оборудования, поддерживающего работу внедряемой системы, а также может значительно снизить затраты на этапе разработки системы.

Также одним из требований, предъявляемых к разрабатываемой системе, является создание удобного, не перегруженного лишней информацией и функционалом пользовательского интерфейса, обеспечивающего легкое восприятие и обработку информации, предоставляющего свободный вывод информации в более привычном для работников предприятия виде, как например, в формате Word документа.

Помимо этого информационная система должна обеспечивать разграничение предоставляемой информации в зависимости от роли пользователя, обладать устойчивостью к ошибкам и к негативным входным данным.

1.4 Выбор средств разработки

Прежде чем приступить к выбору средств разработки, необходимо оценить масштабы будущей ИС, потому как от этого может зависеть, какие технологии необходимо будет использовать для решения поставленной проблемы.

Проведем некоторые математические вычисления. Все используемые при подсчетах данные основаны на показателях филиала ЮУрГУ на текущий момент

времени. Число обучающихся в филиале ЮУрГУ по очной форме обучения – 700 человек, преподавательский состав насчитывает порядка 70 человек, анкета «Преподаватель глазами студента» состоит из 10 вопросов. Учитывая, что в течение года каждый студент изучает не более 20-25 дисциплин, и даже если все дисциплины ведутся разными преподавателями, мы получаем, что БД будет насчитывать 150 тысяч записей. Однако даже это число является завышенным и в реальных условиях число записей не станет превышать сотни тысяч. Для современных систем управления базами данных (далее – СУБД) эта цифра не является большой, а вычислительной мощности современных ЭВМ более чем достаточно для их обработки.

Проведя анализ проблемы можно сделать вывод о том, что решение проблемы автоматизации процесса анкетирования в филиале ЮУрГУ не потребует больших объемов вычислительных мощностей для обработки всего объема данных системы и не сильно ограничивает нас в выборе средств разработки, технологий и СУБД.

На основании этого, разработка автоматизированной системы анкетирования велась с помощью технологии ADO.NET Entity Frameworks. Далее приведен краткий обзор используемых технологий, их особенностей, выбор среды и языка разработки.

1.4.1 Платформа .NET Framework

В 2002 году компания Microsoft выпустила программную платформу, под названием .NET Framework. Основой этой платформы является общезыковая среда исполнения Common Language Runtime (сокращенно – CLR), которая подходит для разных языков программирования. Среда CLR управляет памятью, выполнением потоков, выполнением кода, проверкой безопасности кода, компиляцией и другими системными службами. Эти средства являются внутренними для управляемого кода, который выполняется в среде CLR.

Функциональные возможности CLR доступны в любых языках программирования, использующих эту среду

Основной идеей при разработке .NET Framework являлось обеспечение свободы разработчика за счёт предоставления ему возможности создавать приложения различных типов, способные выполняться на различных типах устройств и в различных средах.

Вторым принципом стала ориентация на системы, работающие под управлением семейства операционных систем Microsoft Windows

Библиотека классов платформы .NET Framework представляет собой коллекцию типов, которые тесно интегрируются со средой CLR. Библиотека классов является объектно-ориентированной; предоставляя типы, из которых управляемый код пользователя может наследовать функции. Это не только упрощает работу с типами .NET Framework, но также уменьшает время, затрачиваемое на изучение новых средств платформы .NET Framework. Кроме того, компоненты независимых производителей можно легко объединять с классами платформы .NET Framework.

Например, в классах коллекций .NET Framework реализуется набор интерфейсов, которые можно использовать для разработки пользовательских классов коллекций. Пользовательские классы коллекций будут без затруднений объединяться с классами .NET Framework.

Как и ожидается от объектно-ориентированной библиотеки классов, типы .NET Framework позволяют решать типовые задачи программирования, включая работу со строками, сбор данных, подключения к серверу БД и доступ к файлам. В дополнение к этим обычным задачам библиотека классов содержит типы, поддерживающие многие специализированные сценарии разработки. Например, можно использовать платформу .NET Framework для разработки следующих типов приложений и служб:

- консольные приложения;

- приложения с графическим интерфейсом пользователя Windows (Windows Forms);
- приложения Windows Presentation Foundation (WPF);
- приложения ASP.NET или web-приложений;
- службы Windows;
- сервисноориентированные приложения с помощью Windows Communication Foundation (WCF);
- приложения поддерживающие бизнес-процессы Windows Workflow Foundation (WF).

1.4.2 Технология ADO.NET Entity Frameworks

ADO.NET Entity Framework (далее – Entity Framework) – объектно-ориентированная технология доступа к данным, является object-relational mapping (ORM) решением для .NET Framework от Microsoft, что означает, что эта технология связывает БД с концепция ми объектно-ориентированных языков программирования, создавая, так называемую виртуальную объектную базу данных. Предоставляет возможность взаимодействия с объектами как посредством LINQ в виде LINQ to Entities, так и с использованием Entity SQL. Для облегчения построения web-решений используется как ADO.NET Data Services (Astoria), так и связка из Windows Communication Foundation и Windows Presentation Foundation, позволяющая строить многоуровневые приложения, реализуя один из шаблонов проектирования MVC, MVP или MVVM.

В качестве одного из проводников между приложением и БД платформы Entity Framework выступает LINQ to Entities. Это альтернативный интерфейс LINQ API, используемый для обращения к БД. Он отделяет сущностную объектную модель данных от физической, вводя логическое отображение между ними. Так, например, схемы реляционных БД не всегда подходят для построения объектно-ориентированных приложений и в результате мы имеем объектную

модель приложения, существенно отличающуюся от логической модели данных, в этом случае используется LINQ to Entities, который использует модель EDM (Entity Data Model).

1.4.3 Объектно-ориентированный язык программирования C#

C# – 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.

Язык программирования C# относится к семье языков с C-подобным синтаксисом, из них его синтаксис наиболее близок к C++ и Java. Язык имеет статическую типизацию, поддерживает полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, свойства, обобщённые типы и методы, итераторы, анонимные функции с поддержкой замыканий, LINQ, исключения, комментарии в формате XML.

Переняв многое от своих предшественников – языков C++, Pascal, Модула, Smalltalk и, в особенности, Java – C#, опираясь на практику их использования, исключает некоторые модели, зарекомендовавшие себя как проблематичные при разработке программных систем, например, C# в отличие от C++ не поддерживает множественное наследование классов (между тем допускается множественное наследование интерфейсов).

Синтаксис C# богат, но при этом прост и удобен в изучении. Характерные фигурные скобки C# мгновенно узнаются всеми, кто знаком с C, C++ или Java. Разработчики, знающие любой из этих языков, обычно очень быстро начинают эффективно работать в C#. Синтаксис C# упрощает многие сложности C++, но при этом предоставляет отсутствующие в Java мощные функции, например обнуляемые типы значений, перечисления, делегаты, лямбда-выражения и прямой

доступ к памяти. C# поддерживает универсальные методы и типы, которые обеспечивают более высокий уровень безопасности и производительности, а также итераторы, позволяющие определять в классах коллекций собственное поведение итерации, которое может легко применить в клиентском коде. Выражения LINQ создают очень удобную языковую конструкцию для строго типизированных запросов.

C# является объектно-ориентированным языком, а значит поддерживает инкапсуляцию, наследование и полиморфизм. Все переменные и методы, включая метод Main, представляющий собой точку входа в приложение, инкапсулируются в определения классов. Класс наследуется непосредственно из одного родительского класса, но может реализовывать любое число интерфейсов. Методы, которые переопределяют виртуальные методы родительского класса, должны содержать ключевое слово `override`, чтобы исключить случайное переопределение. В языке C# структура похожа на облегченный класс: это тип, распределяемый в стеке, реализующий интерфейсы, но не поддерживающий наследование.

Помимо этих основных принципов объектно-ориентированного программирования, C# предлагает ряд инновационных языковых конструкций, упрощающих разработку программных компонентов.

- Инкапсулированные сигнатуры методов, именуемые делегатами, которые позволяют реализовать типобезопасные уведомления о событиях.
- Свойства, выполняющие функцию акцессоров для закрытых переменных-членов.
- Атрибуты, предоставляющие декларативные метаданные о типах во время выполнения.
- Внутрострочные комментарии для XML-документации.
- LINQ для создания запросов к различным источникам данных.

1.4.4 Среда разработки и интерфейс приложения

Microsoft Visual Studio – линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также сайты, web-приложения, web-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Silverlight.

Visual Studio включает в себя редактор исходного кода с поддержкой технологии IntelliSense и возможностью простейшего рефакторинга кода. Встроенный отладчик может работать как отладчик уровня исходного кода, так и отладчик машинного уровня. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения, web-редактор, дизайнер классов и дизайнер схемы БД. Visual Studio позволяет создавать и подключать сторонние дополнения (плагины) для расширения функциональности практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (как, например, Subversion и Visual SourceSafe), добавление новых наборов инструментов (например, для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования) или инструментов для прочих аспектов процесса разработки программного обеспечения (например, клиент Team Explorer для работы с Team Foundation Server).

Visual Studio Community 2015 – это свободно распространяемая полнофункциональная интегрированная среда разработки с мощными, эффективными возможностями для кодирования, инструментами кроссплатформенных разработок мобильных приложений для Windows, iOS и Android, web- и облачных приложений, а также доступом к тысячам расширений.

Разработка системы анкетирования велась именно с помощью данной версии продукта, потому как этот выпуск Visual Studio свободно доступен для отдельных разработчиков, для разработки проектов с открытым исходным кодом, академических исследований, образования и небольших групп специалистов.

Одним из требований информационной является простой и удобный интерфейс. Графический интерфейс пользователя информационной системы будет представлен в виде оконного приложения, разработанного с помощью Windows Forms (далее – WinForms). WinForms – интерфейс программирования приложений (API), отвечающий за графический интерфейс пользователя и являющийся частью Microsoft .NET Framework. Данный интерфейс упрощает доступ к элементам интерфейса Microsoft Windows за счет создания обёртки для существующего Win32 API в управляемом коде. Причём управляемый код – классы, реализующие API для WinForms, не зависят от языка разработки. То есть программист одинаково может использовать WinForms как при написании ПО на C#, C++, так и на VB.Net, J# и др.

1.5 Выбор СУБД

В большинстве случаев при создании собственной ИС невозможно обойтись без использования БД. Поэтому выбор СУБД является одним из важных этапов при разработке автоматизированной системы анкетирования обучающихся университета. Выбранный программный продукт должен удовлетворять как текущим, так и будущим потребностям учебного заведения, при этом следует учитывать затраты на разработку и настройку необходимого программного обеспечения, а также обучение персонала работе с ним.

Наиболее правильный подход при выборе СУБД основан на оценке того, в какой мере существующие системы удовлетворяют основным требованиям создаваемой информационной системы.

Существует несколько критериев выбора СУБД:

- моделирование данных;
- особенности архитектуры и функциональные возможности;
- контроль работы системы;
- особенности разработки приложений;
- производительность;
- надежность;
- требования к рабочей среде;
- смешанные критерии.

При разработке информационной системы для сбора и анализа данных анкетирования большое внимание следует уделить обеспечению одновременной работы нескольких пользователей, что подразумевает под собой использование большого числа вычислительных машин и одновременный доступ к хранилищу данных. Данные проблемы отлично решают СУБД, построенные на архитектуре клиент сервер.

Клиент-серверные системы управления базами данных имеют следующие положительные стороны:

- массивы данных не перекачиваются по сети от сервера БД на компьютер пользователя;
- требования к пропускной способности сети понижаются. Это делает возможным одновременную работу большого числа пользователей с большими объемами данных;
- обработка данных осуществляется на сервере БД, а не в компьютере пользователей. Что позволяет использовать более простые, а значит, дешевые компьютеры на клиентских местах;
- блокировки (захвата) данных одним пользователем не происходит.
- обеспечивается доступ пользователя не к целому файлу, а только к тем данным из него, с которыми пользователь имеет право работать.

К недостаткам стоит отнести повышенные требования к серверу. Так неработоспособность сервера может сделать неработоспособной всю вычислительную сеть. Также обслуживание такой сети требует отдельного специалиста – системного администратора.

Исходя из анализа общих требований предъявляемых к надежности и производительности, лидируют продукты Oracle, Microsoft SQL Server, MySQL. Только они могут обеспечить сетевой доступ, надлежащую масштабируемость будущей системы, позволяют на высоком уровне создавать сложные распределенные архитектуры.

MySQL — свободная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация Oracle, имеющая на данный момент права на торговую марку. Продукт распространяется как под GNU General Public License, так и под собственной коммерческой лицензией. Помимо этого разработчики создают функциональность по заказу лицензионных пользователей, именно благодаря такому заказу почти в самых ранних версиях появился механизм репликации.

MySQL является решением для малых и средних приложений. Входит в состав серверов WAMP, AppServ, LAMP и в портативные сборки серверов Денвер, XAMPP. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей. Более того, СУБД MySQL поставляется со специальным типом таблиц EXAMPLE, демонстрирующим принципы создания новых типов таблиц. Благодаря открытой

архитектуре и GPL-лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц.

MySQL является наиболее приспособленной для применения в среде web СУБД. При этом она стала незыблемым стандартом в области СУБД для web, в которой развиваются возможности для использования ее в любых критичных бизнес-приложениях, что создает конкуренцию на равных с СУБД таких производителей, как Oracle, IBM, Microsoft и Sybase.

Основные преимущества MySQL:

- многопоточность, поддержка нескольких одновременных запросов;
- оптимизация связей с присоединением многих данных за один проход;
- записи фиксированной и переменной длины;
- ODBC драйвер;
- гибкая система привилегий и паролей;
- гибкая поддержка форматов чисел, строк переменной длины и меток времени;
- интерфейс с языками C и Perl, PHP;
- быстрая работа, масштабируемость;
- совместимость с ANSI SQL;
- бесплатна;
- хорошая поддержка со стороны провайдеров услуг хостинга.

Выводы по разделу один:

В данном разделе мы провели анализ предметной области, включающий в себя, описание и структурный анализ, а так же сформировали цели и задачи для разработки данной системы. Рассмотрели возможности технологии Entity Frameworks, привели обоснование выбора среды разработки ПО и языка программирования.

2 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА БД И ПРИЛОЖЕНИЯ ДЛЯ АНКЕТИРОВАНИЯ

2.1 Разработка БД

Многолетним и общим подходом к разработке является подход, при котором построение приложения или службы представляет собой его разделение на три части: концептуальную, логическую и физическую модель. Концептуальная модель домена определяет сущности и связи в моделируемой системе. Логическая модель для реляционной БД обеспечивает нормализацию сущностей и связей в целях создания таблиц с ограничениями внешнего ключа. В физической модели учитываются возможности конкретной системы обработки данных путем определения зависящих от ядра БД подробных сведений о хранении данных, которые касаются секционирования и индексирования.

Entity Framework (далее – EF) предполагает три возможных способа взаимодействия с БД:

- Database first: EF создаёт набор классов, которые отражают модель конкретной БД;
- Model first: сначала разработчик создаёт модель БД, по которой затем EF создаёт реальную БД на сервере;
- Code first: разработчик создаёт класс модели данных, которые будут храниться в БД, а затем EF по этой модели генерирует БД и её таблицы.

При разработке данной ИС БД выступает в качестве хранилища данных о преподавателях, вопросах и данных самого анкетирования, основная логика обработки данных будет заключена в самом приложении, поэтому при разработке приложения будет использован подход Code First. Также плюсом данного подхода является простота внедрения готового программного продукта, так как создание БД и заполнения ее необходимыми для начала работы данными происходит автоматически во время работы программы.

Для того, чтобы было удобнее определиться с объектами, проведем анализ предметной области. В разделе один были выявлены некоторые сущности, такие как Преподаватели и Вопросы анкеты. Теперь определим все сущности предметной области. В качестве основного атрибута всех сущностей будет выступать уникальный идентификатор (далее – Id):

- Кафедра;
- Преподаватель;
- Вопрос анкеты;
- Особое мнение.

Не смотря на то, что Особое мнение (далее – комментарий) является одним из вопросов анкеты, он считается не обязательным и имеет строковый тип записи, в отличии от других вопросов, поэтому выделяется как отдельная сущность.

Определим связи между сущностями:

1. На Кафедре (1) числится хотя бы один (М) Преподаватели;
2. Преподаватели (–М) оцениваются по всем (М) Вопросам анкеты;
3. Комментарий (–М) оставляется на некоторых (1) Преподавателя.

Обозначение (–) указывает на необязательный класс принадлежности сущности.

На основе составленных связей между сущностями построим следующие отношения:

1. Кафедра(Id кафедры);
Преподаватель(Id преподавателя, Id кафедры);
2. Преподаватель(Id преподавателя);
Вопрос анкеты(Id вопроса);
Ответ (Id преподавателя, Id вопроса);
3. Комментарий(Id комментария, Id преподавателя);
Преподаватель(Id преподавателя).

После добавления не ключевых атрибутов получаем модель БД, состоящей из следующих сущностей:

Кафедра

- Id;
- Наименование.

Преподаватель

- Id;
- ФИО;
- Id кафедры.

Вопрос анкеты

- Id;
- Наименование вопроса.

Ответ

- Id преподавателя;
- Id вопроса;
- Значение.

Комментарий

- Id;
- Id преподавателя;
- Значение.

Подход Code First позволяет работать с сущностями предметной области как с объектами. Так каждая сущность будет представлена в виде класса, атрибутами которого будут являться атрибуты сущностей. А связь один-ко-многим реализуется следующим образом: одна модель хранит ссылку на один объект другой модели, а вторая модель ссылается на коллекцию объектов первой модели. Так, подход Code First не позволяет реализовать связь многие ко многим с включением дополнительных атрибутов, поэтому необходимо выделить Ответы как отдельную сущность и на основе этого пересмотреть связь многие-ко-многим как две связи один-ко-многим. Во всем остальном определенные сущности будут представлены в виде объектов без изменений.

Реализация объектов предметной области в коде программы представлена на рисунках 2.1 и 2.2.

```
public class Kafedra
{
    ссылка: 3
    public int Id { get; set; }
    ссылка: 8
    public string KafName { get; set; }

    ссылка: 1
    public ICollection<Prepodavatel> Prepodavatels { get; set; }
    ссылка: 5
    public Kafedra()
    {
        Prepodavatels = new List<Prepodavatel>();
    }
}
ссылка: 14
public class Prepodavatel
{
    ссылка: 4
    public int Id { get; set; }
    ссылка: 7
    public string PrepName { get; set; }

    ссылка: 6
    public int KafedraId { get; set; }
    ссылка: 2
    public Kafedra Kafedra { get; set; }
    ссылка: 1
    public ICollection<Otvets> Otvets { get; set; }
    ссылка: 2
    public Prepodavatel()
    {
        Otvets = new List<Otvets>();
    }
}
ссылка: 24
public class Vopros
{
    ссылка: 0
    public int Id { get; set; }
    ссылка: 11
    public string VoprString { get; set; }
    ссылка: 1
    public ICollection<Otvets> Otvets { get; set; }
    ссылка: 10
    public Vopros()
    {
        Otvets = new List<Otvets>();
    }
}
--
```

Рисунок 2.1 – Реализация БД в подходе Code First


```

ссылка: 29
public class Otvet
{
    ссылка: 0
    public int Id { get; set; }
    ссылка: 10
    public int PrepodavatelId { get; set; }
    ссылка: 12
    public int VoprosId { get; set; }
    ссылка: 11
    public float OtvetValue { get; set; }
    ссылка: 4
    public Prepodavatel Prepodavatel { get; set; }
    ссылка: 0
    public Vopros Vopros { get; set; }
}
ссылка: 3
public class OtvetComentary
{
    ссылка: 0
    public int Id { get; set; }
    ссылка: 2
    public int PrepodavatelId { get; set; }
    ссылка: 3
    public string Comentary { get; set; }
    ссылка: 0
    public Prepodavatel Prepodavatel { get; set; }
}

```

Рисунок 2.2 – Реализация БД в подходе Code First

На основе каждого класса будет создана таблица внутри БД. Для взаимодействия классов с таблицами БД используется контекст (класс производный от DbContext) и набор данных DbSet. Реализация класса контекста представлена на рисунке 2.3.

```

[DbConfigurationType(typeof(MySqlEFConfiguration))]
ссылка: 12
class UniverContext : DbContext
{
    ссылка: 3
    public UniverContext()
        : base("UniverContext")
    { }
    ссылка: 3
    public UniverContext(string connString)
        : base(connString)
    { }
    ссылка: 12
    public DbSet<Prepodavatel> Prepodavatels { get; set; }
    ссылка: 9
    public DbSet<Kafedra> Kafedras { get; set; }
    ссылка: 12
    public DbSet<Vopros> Vopross { get; set; }
    ссылка: 15
    public DbSet<Otvets> Otvets { get; set; }
    ссылка: 4
    public DbSet<OtvetsComentary> OtvetsComentarys { get; set; }
}

```

Рисунок 2.3 – Контекст данных

Класс контекста содержит два конструктора базового класса. При вызове конструктора без параметров подключение к серверу БД будет задаваться с помощью строки подключения, заданной в конфигурационном файле. Пример строк подключения, описанный в конфигурационном файле приложения, показан на рисунке 2.4.

```

<connectionStrings>
  <add name="UniverContext" connectionString="Server=localhost; Port=3306;
    Uid=newuser; Pwd=12345; Database=UniverDB3;" providerName="MySQL.Data.MySqlClient" />
</connectionStrings>

```

Рисунок 2.4 – Строка подключения

При вызове конструктора в качестве параметра можно передать строку подключения, по которому будет осуществлено подключение к БД.

При первом же обращении к объекту класса во время работы приложения автоматически создаются таблицы и связи между ними. Физическая модель БД представлена на рисунке 2.5.

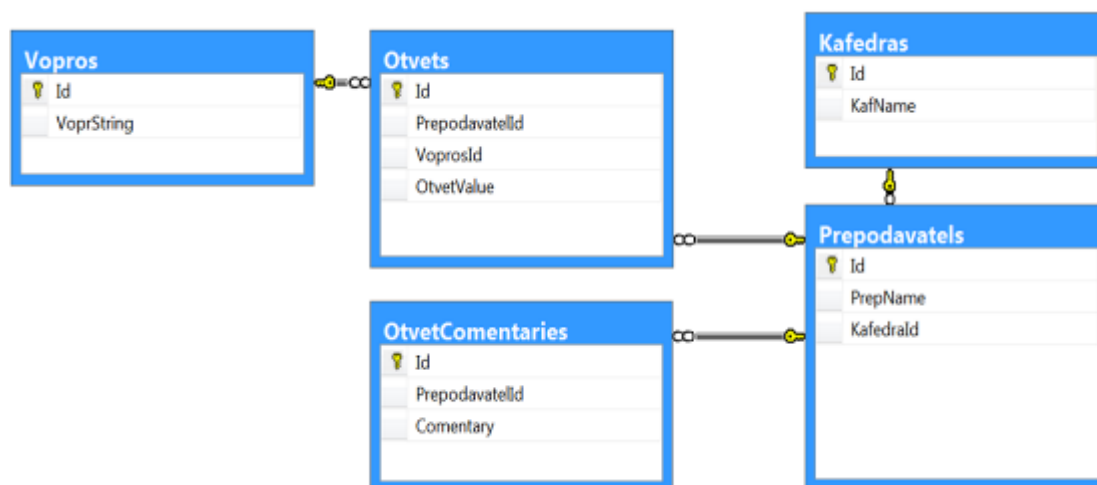


Рисунок 2.5 – Физическая модель БД

2.2 Разработка приложения

Одним из требований к приложению является разделения предоставляемого функционала в зависимости от роли пользователя. Было выделено две роли пользователя:

- Студент: пользователю предоставляется форма анкетирования «Преподаватель глазами студента»;
- Администратор: предоставляются результаты анкетирования, возможность составления отчетов, удаление данных анкетирования, редактирование списка преподавателей.

Для осуществления разделения функционала в приложении предусмотрена авторизация пользователей.

2.2.1 Вход в систему

Первое что видит пользователь при запуске приложения это окно «Вход в систему», оно представлено на рисунке 2.6

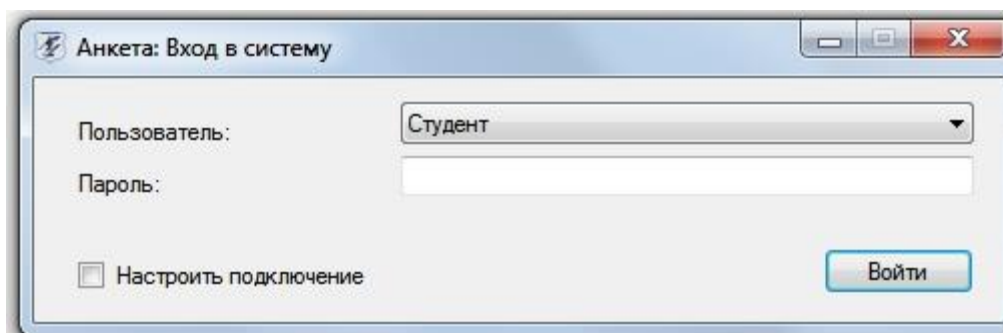


Рисунок 2.6 – Вход в систему

Здесь пользователю предоставляется возможность выбора режима работы приложения. Также в этом окне можно настроить подключение к серверу БД отметив пункт Настроить подключение (рисунок 2.7).

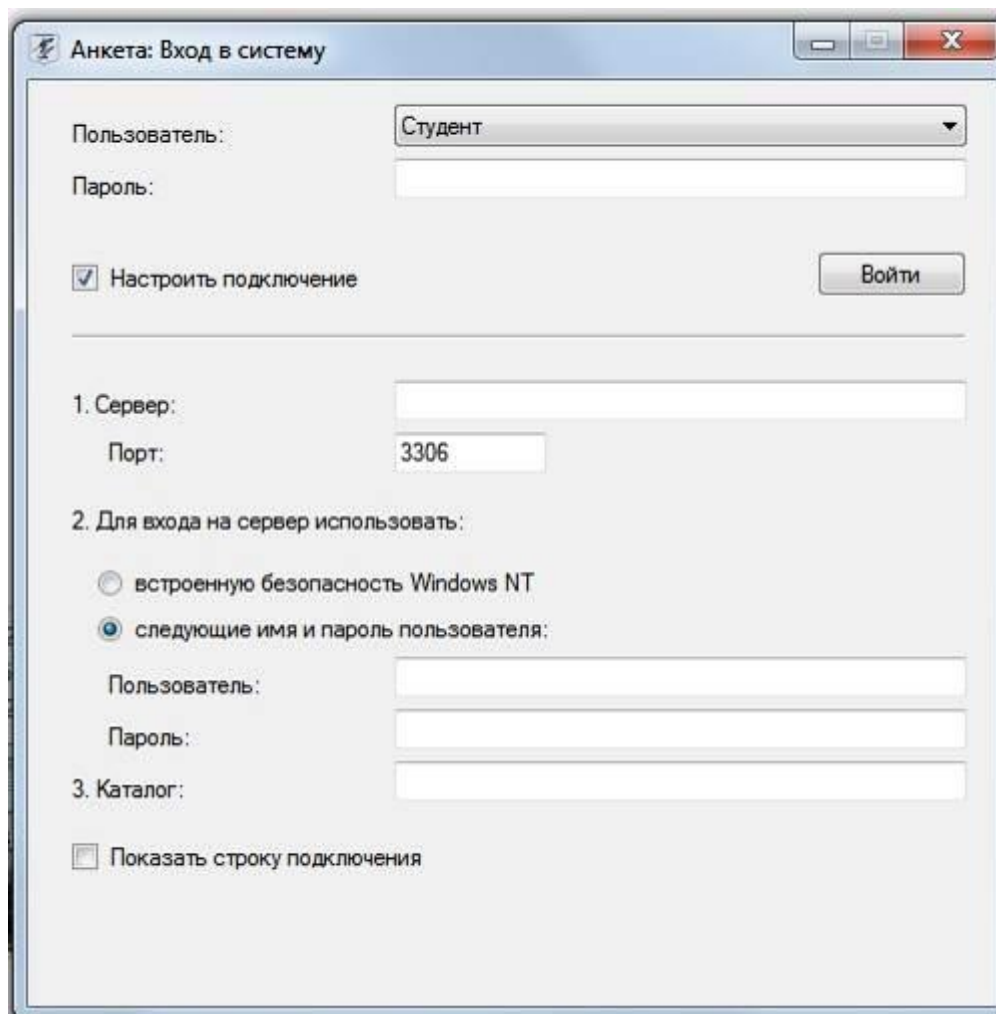


Рисунок 2.7 – Настройка подключения

Приложение разрабатывалось с помощью интерфейса WinForms. При реализации окна входа в систему были использованы следующие компоненты:

- Button: кнопка подтверждения входа в систему;
- ComboBox: предоставляет пользователю выбор режима работы приложения;
- TextBox: текстовое поле ввода данных от пользователя;
- CheckBox: предоставляет пользователю выбор того или иного параметра;
- RadioButton: элемент, предоставляющий пользователю выбор условий;
- Label: надпись-подсказка для пользователя.

Теперь рассмотрим логику события, которое вызывается нажатием на кнопку. Для предотвращения ошибок, которые могут возникнуть в случае подключения к БД, используется конструкция перехвата ошибок `try..catch`.

Сначала проверяется, выбран ли параметр настройки подключения. Если да, то будет сформирована новая строка подключения, по которому будет осуществлено подключение. Строка формируется на основе данных, которые пользователь ввел в соответствующие поля ввода.

Далее выполняется проверка подключения к БД. Для этого на сервер БД отправляется простой SQL запрос. Если подключение было настроено неверно, данная процедура завершится ошибкой.

Затем в зависимости от выбранного режима работы пользователем, приложение откроет ту или иную форму. Если пользователь выбрал режим работы «Администратор», прежде чем вызвать конструктор формы проверится правильность введенного пароля.

2.2.2 Форма Студент

Конструктор формы, предназначенной для сбора данных анкетирования, представлена на рисунке 2.8

ФИО преподавателя	Вопрос1	Вопрос2	Вопрос3	Вопрос4	Вопрос5	Вопрос6	Вопрос7	Вопрос8	Вопрос9	Commentary
-------------------	---------	---------	---------	---------	---------	---------	---------	---------	---------	------------

Критерии оценивания:
1 - недостаточно
2 - достаточно
3 - отлично

Рисунок 2.8 – Форма Студент

При разработке формы студента, помимо описанных выше компонентов, был использован компонент DataGridView. Этот компонент отображается пользователю в виде таблицы, в которой он может вводить данные.

Для отображения таблицы в нужном виде и для хранения данных введенных пользователем перед сохранением их в БД был разработан класс AnketaPresent, код которого представлен на рисунке 2.9.

```

public class AnketaPresent
{
    ссылка: 11
    public int PrepodavatelId { get; set; }
    ссылка: 1
    public string PrepName { get; set; }
    ссылка: 1
    public string Vopros1 { get; set; }
    ссылка: 1
    public string Vopros2 { get; set; }
    ссылка: 1
    public string Vopros3 { get; set; }
    ссылка: 1
    public string Vopros4 { get; set; }
    ссылка: 1
    public string Vopros5 { get; set; }
    ссылка: 1
    public string Vopros6 { get; set; }
    ссылка: 1
    public string Vopros7 { get; set; }
    ссылка: 1
    public string Vopros8 { get; set; }
    ссылка: 1
    public string Vopros9 { get; set; }
    ссылка: 1
    public string Comentary { get; set; }
}

```

Рисунок 2.9 – Класс представления анкеты пользователю

У компонент DataGridView в свойстве DataSource указана ссылка на класс представления анкеты. Также были настроены столбцы следующим образом: столбец PrepodavatelID был удален, а у столбца PrepName установлено поведение только для чтения.

Так как один студент может оценивать нескольких преподавателей сразу, и у разных студентов число преподавателей может быть различно, данные будут храниться в типизированном списке List<>, который имеет название otvetiStudentaList. При добавлении или удалении из этого списка преподавателя компоненту DataGridView в качестве источника данных передается этот измененный список. Теперь все данные, который вводит пользователь в таблицу, сохраняются в переменной otvetiStudentaList и хранятся там до завершения процесса анкетирования.

Список всех преподавателей, которых можно оценить в анкете, хранится в компоненте ComboBox. При загрузке формы происходит заполнение списка

значений из БД. Также при загрузке формы заголовки столбцов компонента DataGridView переименовываются в соответствии со значением вопроса из БД.

Сохранение данных анкетирования в БД происходит после нажатия на кнопку «Готово». При этом проверяется:

- все ли необходимые ячейки заполнены;
- являются ли введенные пользователем данные целыми числами;
- удовлетворяют ли введенные значения критериям анкетирования.

В случае выполнения всех этих условий данные из переменной `otvetiStudentaList` преобразуются из строкового типа в числовой и сохраняются в БД, в чем удостоверяется пользователь, после этого приложение завершает работу. Если по какой либо причине сохранение данных в БД не возможно, пользователю выдается соответствующее сообщение.

2.2.2 Форма Администратор

Как было отмечено ранее, подход Code First подразумевает под собой автоматическое создание БД. Для нормального функционирования системы и уменьшения объема работы по развертыванию приложения на предприятии, необходимо разработать автоматическое заполнение БД всеми необходимыми данными, такими как наименование вопросов анкетирования, названия кафедр и список преподавателей. Для этого при загрузке формы администратора выполняется проверка, являются ли соответствующие таблицы пустыми.

Данные о преподавателях хранятся в ресурсах проекта в виде строк [ФИО_преподавателя]/[Код_кафедры]. При заполнении таблицы преподавателей данные построчно считываются из ресурсов, строка разбивается соответственно на ФИО преподавателя и на код кафедры, после чего данные заносятся в таблицу БД. В связи с тем, что данные о наименовании вопросов и названий кафедр гораздо меньше по объему, чем данные о преподавателях, они добавляются в БД из кода программы.

Все данные, от фамилии преподавателя до значения ответа студента хранятся в БД. Для того, чтобы минимизировать число запросов в БД и тем самым снизить нагрузку на сеть, эффективнее хранить данные в переменной. В качестве результатов анкетирования для каждого преподавателя необходимо хранить Id, ФИО, средний балл, количество респондентов и список комментариев. Также уместно указать Id кафедры и ее наименование. Таким образом, класс, описывающий тип данных, представлен на рисунке 2.10.

```
public class AnketaSQLResult
{
    ссылка: 2
    public int PrepId { get; set; }
    ссылка: 5
    public string PrepName { get; set; }
    ссылка: 10
    public int KafId { get; set; }
    ссылка: 6
    public string KafName { get; set; }
    ссылка: 1
    public string KafShortName { get; set; }
    ссылка: 5
    public int RespCount { get; set; }
    ссылка: 16
    public double otvetValueAvg { get; set; }
    ссылка: 6
    public List<string> Comentary { get; set; }
    ссылка: 2
    public AnketaSQLResult()
    {
        Comentary = new List<string>();
    }
}
```

Рисунок 2.10 – Класс хранения результатов запроса из БД

Форма администратора предоставляет пользователю следующий функционал:

- просмотр результатов анкетирования;
- сохранение полного отчета;
- настройка отчета и сохранение его с заданными настройками;
- редактирование списка преподавателей.

Для разделения рабочей области окна приложения между разными функциями, предоставляемыми формой администратора, используется компонент TabControl.

Данные анкетирования представлены пользователю в виде диаграмм с помощью компонента Chart. Для сохранения отчетов в Word-документ используется библиотека работы с пакетом приложений Office. Interop.Word.

Окно настройки формируемого отчета представлено на рисунке 2.11.

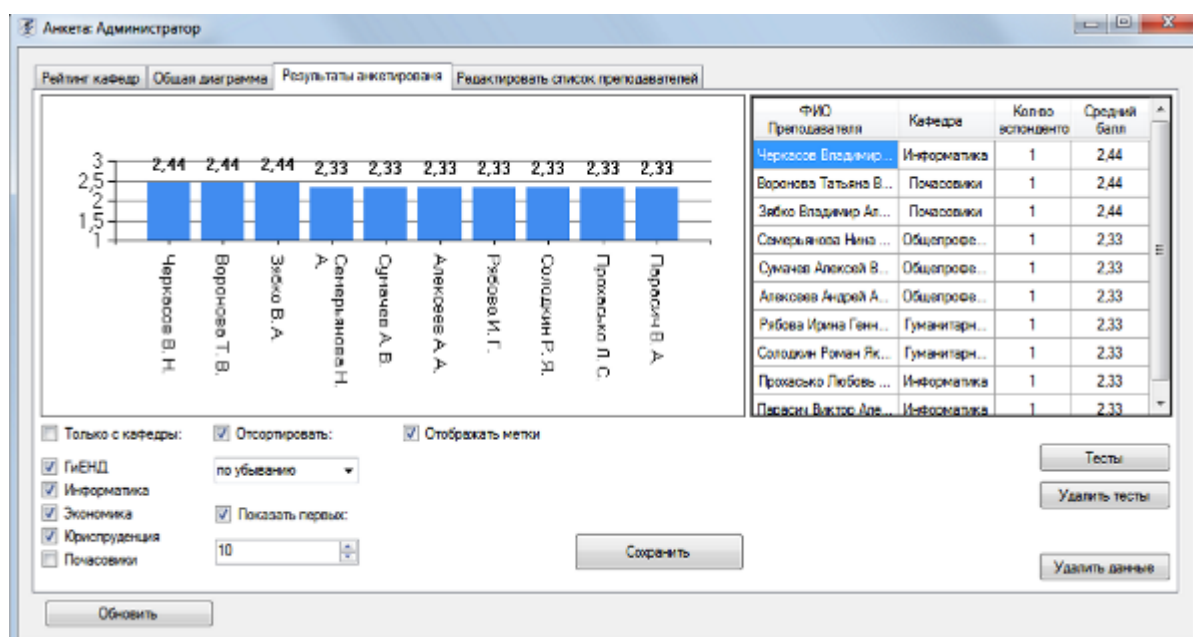


Рисунок 2.11 – Настройка отображения результатов анкетирования

Настройка формируемого отчета располагается на отдельной вкладке компонента TabControl. Настройка данных анкетирования со стороны осуществляется с помощью компонентов ComboBox и CheckBox. При изменении состояний этих компонентов вызывается функция настройки отображаемых данных. Настройка осуществляется с помощью языка LINQ. LINQ позволяет использовать SQL-подобный синтаксис непосредственно в коде программы, может применяться не только для формирования запросов БД, но и при работе с объектами.

Для редактирования списка преподавателей пользователю предлагается интерфейс, представленный на рисунке 2.12.

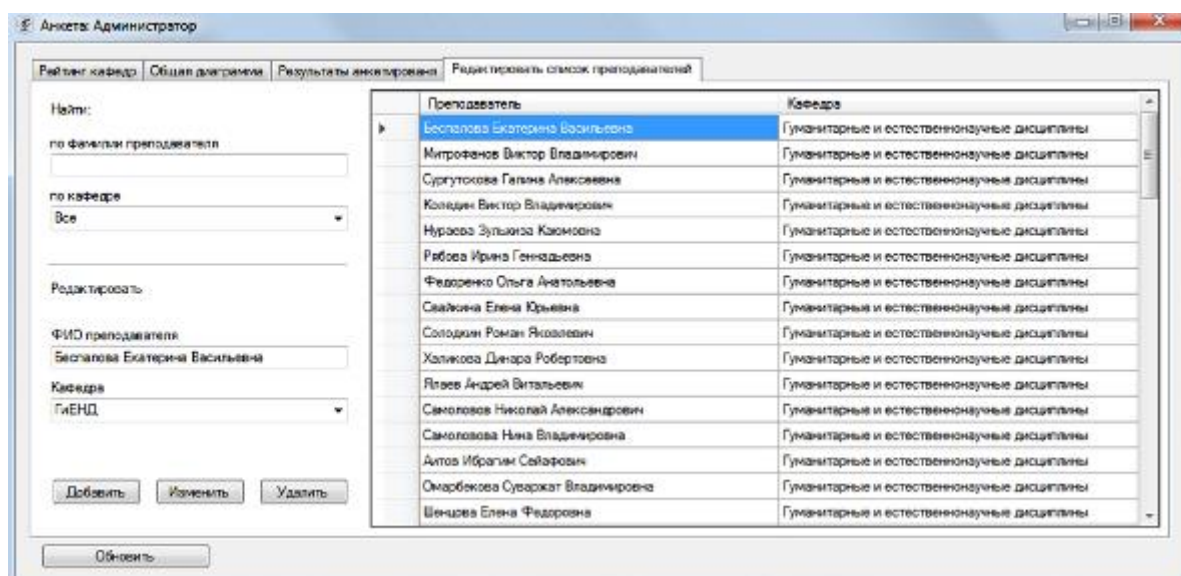


Рисунок 2.12 – Редактирования списка преподавателей

Технология EF позволяет легко реализовать редактирование записей в БД. Работа ведется непосредственно с объектами, представляющими таблицы БД, редактируя значения атрибутов, всю остальную работу по составлению запросов к базе данных EF формирует на основе написанного кода.

Выводы по разделу два:

В данном разделе на основании результатов анализа предметной области была реализована БД. Также в соответствии с требованиями к системе разработано приложение для сбора и обработки данных анкетирования на основании выбранных средств разработки.

3 ОРГАНИЗАЦИОННО-ЭКОНОМИЧЕСКИЙ РАЗДЕЛ

Внедрение автоматизированных систем обеспечивает удобство в работе, рациональную организацию производства и снижение психологических нагрузок. Также снижаются физиологические нагрузки, т.к. с внедрением соответствующего программного обеспечения время, затраченное на эту же работу, существенно уменьшается. Это положительно влияет на работоспособность работников, т.к. приводит к уменьшению количества обрабатываемой информации. Кроме того у работников появляется дополнительное время на решение других задач.

Эффективность от внедрения системы заключается в автоматизировании процессов проведения анкетирования среди студентов:

- анкетирование студентов;
- анализа полученных данных.

Снижение трудовых затрат позволит уменьшить и финансовые затраты, что приведет к общему увеличению производительности.

В данном разделе будет рассчитан экономический эффект от использования автоматизированной системы анкетирования.

Основной задачей этой главы является определение величины затрат на проведение исследований, себестоимость для определения экономического эффекта от использования в общественном производстве основных и сопутствующих результатов, получаемых при решении поставленной технической задачи в данной работе. Оценка эффективности принятого научно-технического решения должна учитывать все необходимые расходы и затраты, для этого требуется провести ряд необходимых расчетов.

3.1 Расходы на приобретение, содержание и эксплуатацию программного и аппаратного обеспечения

3.1.1 Расходы на разработку

Все программное обеспечение, которое использовалось при разработке, такие как Visual Studio 2015 и MySQL имеют бесплатную лицензию. Также, установка и эксплуатация информационной системы анкетирования не требует установки дополнительного программного обеспечения, так как все необходимое оборудование для работы системы уже имеется в университете.

Таким образом, при расчетах затрат на разработку и эксплуатацию системы анкетирования, расход на программное обеспечение принимается равным нулю.

Стоимость объектов по элементам затрат основана на списке цен в сетевых магазинах, и представлена в таблице 3.1.

Таблица 3.1 – Расходы на материалы

Наименование	Количество	Цена рублей
Бумага SvetoCopy A4	1	204
USB Flash Kingston DataTraveler DT50 8 ГБ	1	510
Диск CD-R 700Mb 52x SlimCase VS	1	44
Итого		758

Затраты на электроэнергию находятся исходя из продолжительности периода разработки ПО, количества кВт/ч, затраченных на проектирование ПО и тарифа за 1 кВт/ч. Тариф по городу Нижневартовску для физических лиц составляет 2,73 руб. за кВт/ч (согласно данным, представленным на официальном сайте ООО «Нижневартовская энергосбытовая компания»). Затраты отражены в таблице 3.2.

Таблица 3.2 – Затраты на электроэнергию

Наименование работы	Установленная мощность, кВт	Стоимость 1кВт в час (руб.)	Количество часов работы	Общая стоимость рублей
Разработка системы	0,045	2,73	155	32,64
Итого				32,64

Затраты на амортизацию оборудования проводятся за период их использования, т.е. за период создания информационной системы.

Денежное выражение амортизации является амортизационным отчислением, которое входит в текущие затраты.

Величина амортизационных отчислений определяется на основе норм амортизации.

Норма амортизации – это установленный размер амортизационных отчислений на полное восстановление, выраженное в процентах. Норма амортизации устанавливается на основе экономически целесообразного срока службы и должна обеспечить возмещение износа основных средств к моменту возможного их морального и физического износа и создать экономическую основу для замены.

Амортизационные отчисления, приходящиеся на 1 час работы системы, рассчитывается по формуле (1).

$$Aч = \Phi_{\text{перв}} * \frac{a}{F_d}, \quad (1)$$

где $\Phi_{\text{перв}}$ – первоначальная стоимость системы или отдельных элементов;

a – норма амортизации (0.2);

F_d – фонд времени работы за год (2500 часов).

Расчет амортизационных отчислений представлен в таблице 3.3

Таблица 3.3 – Расчет амортизационных отчислений

п/п	Элемент КТС	Фперв	Фд	Ач	Количество часов работы	Общая стоимость (руб.)
	Стационарный компьютер	55 370	2500	4,43	155	686,65
ИТОГО						686,65

Просуммировав расчеты, мы получили расходы на приобретение, содержание и эксплуатацию программного обеспечения равными 1477,29 руб.

3.1.2 Затраты на заработную плату

Для расчета расходов на заработную плату необходимо умножить среднюю часовую ставку программиста на трудоемкость работы, чел/час по каждому из этапов разработки системы.

Средняя часовая ставка взята по формуле (2):

$$Зч = (Зм)/168, \quad (2)$$

где $Зч$ – средняя часовая ставка программиста;

$Зм$ – средняя месячная ставка программиста (30000 рублей).

$$Зч = 30000 / 168 = 180 \text{ рублей.}$$

$Зч = 250$ рублей – среднечасовая ставка руководителя выпускной квалификационной работы и консультанта по экономике и БЖД.

Исходя из полученных данных, можно вычислить заработную плату по всем этапам разработки, результат в таблице 3.4.

Таблица 3.4 – Расчет основной заработной платы

п/п	Содержание работы	Трудоемкость работы, чел/час	Основная заработная плата (руб.)
1.	Анализ предметной области	8	1440
2.	Постановка задачи	8	1440
3.	Установка и настройка программных средств разработки	4	720
4.	Создание моделей базы данных	32	5760
5.	Разработка клиентского приложения	40	7200
6.	Тестирование и отладка системы	32	5760
7.	Обучение персонала	8	1440
8.	Внедрение	4	720
9.	Документирование	11	1980
10.	Руководство ВКР (руководитель и консультанты)	8	2000
11.	Итого	155	28460

3.1.3 Расчет затрат на дополнительную заработную плату

Дополнительную заработную плату разработчиков определяют в процентах от итоговой суммы основной заработной платы (15 %).

$$\text{ЗП доп.} = 28460 * 0,15 = 4269 \text{ (руб.)}$$

3.1.4 Отчисления на социальные нужды

Единый социальный налог рассчитывают в процентах от суммы основной и дополнительной заработных плат, в пенсионный фонд, в фонд социального страхования и медицинское страхование. На 2017 год данный процент составляет 30%, социальные вычеты рассчитываются по формуле (3).

$$СВ = 30\% \cdot (ЗП_{осн} + ЗП_{доп}), \quad (3)$$

где ЗП_{осн} – основная заработная плата;

ЗП_{доп} – дополнительная заработная плата.

Социальные вычеты = 0,30 * (28460 + 4269) = 9818,7 (руб.)

3.1.5 Общая смета затрат на внедрение системы

Теперь найдем итоговую стоимость системы анкетирования. Общая смета затрат на разработку системы представлена в таблице 3.5

Таблица 3.5 – Общая смета затрат

п/п	Элементы затрат	Сумма, руб.
1	Приобретение, содержание и эксплуатация программного и аппаратного обеспечения	1477,29
2	Затраты на основную заработную плату	28460
3	Затраты на дополнительную заработную плату	4269
4	Отчисления на социальные нужды	9818,7
ИТОГО		44024,99

3.2 Расчет экономического эффекта

Главный экономический эффект от внедрения средств автоматизации заключается в улучшении экономических и хозяйственных показателей работы предприятия, в первую очередь за счет повышения оперативности управления и снижения трудозатрат на реализацию процесса управления, то есть сокращения расходов на управление. Для большинства предприятий экономический эффект выступает в виде экономии трудовых и финансовых ресурсов, получаемой от:

- снижение трудозатрат на сбор информации
- снижения трудоемкости расчетов;
- снижение трудозатрат на поиск и подготовку документов;
- экономии на расходных материалах (бумага, картриджи);

Критерием эффективности создания и внедрения новых средств автоматизации является ожидаемый экономический эффект.

3.2.1 Расчет экономии за счет увеличения производительности труда пользователя

Если пользователь при экономии i -вида с применением программы экономит T_i , часов, то повышение производительности труда P_i (в %) определяется по формуле (4):

$$P_i = \left(\frac{T_i}{F_i - T_i} \right) * 100, \quad (4)$$

где F_i – время, которое планировалось пользователем для выполнения работы i -вида до внедрения программы.

В таблице 3.6 приведен расчет производительности труда от внедрения информационной системы.

Таблица 3.6 – Расчет повышения производительности труда

п/п	Вид работ	До автоматизации (час) F	После автоматизации (час)	Экономия времени (час) T	Повышение производительности труда (%) P
1	Подготовка и проведение анкетирования обучающихся	24	14	10	71,4
2	Обработка данных анкетирования	16	0,1	15,9	15900

Как мы видим из приведенных расчетов, внедрение существенно повышает производительность труда на этапе обработки данных анкетирования, что в свою очередь ведет к снижению трудозатрат работника, отвечающего за расчет результатов анкетирования.

3.2.2 Расчет экономического эффекта

При расчете экономического эффекта внедрения системы автоматизации анкетирования, нужно составить стоимость каждого процесса до автоматизации и после. Также необходимо учесть значение средней стоимости часа работника выполняющего операции

Процесс анкетирования состоит из двух основных процессов: сбор данных и их обработка. Таким образом, средняя заработная стоимость часа преподавателя, осуществляющего подготовку и анкетирование обучающихся, составляет:

$Z_{ч_п} = 150$ рублей.

В обработке результатов анкетирования принимает участие системный администратор, поэтому при расчетах необходимо учитывать его заработную плату.

$Z_{ч_а} = 240$ рублей – среднечасовая ставка системного администратора.

Далее составить таблицу 3.7 бизнес процессов с указанием часов до и после автоматизации и расчетом согласно стоимости часа.

Таблица 3.7 – Расчет стоимости процессов до и после автоматизации

п/п	Показатель	Количество часов до внедрения	Количество часов после внедрения	Стоимость процесса до автоматизации, руб.	Стоимость процесса после автоматизации, руб.
1	Подготовка к анкетированию	8	0	1216	0
2	Анкетирование студентов	16	14	2432	2128
3	Обработка результатов	16	0,1	3840	24
	ИТОГО	40	14,1	7488	2152

Также стоит учитывать, что до автоматизации анкеты были в печатном виде, следовательно, при оценке стоимости процесса анкетирования до автоматизации, необходимо также учесть и материальные расходы. Материальные затраты на проведение анкетирования до автоматизации представлены в таблице 3.8

Таблица 3.8 – Расчет материальных затрат процесса до автоматизации

Наименование	Количество	Цена рублей
Бумага SvetoCopy A4	2	408
Чернила Cactus CS-I-PG40	1	140
Итого		548

Тогда стоимость процесса до автоматизации составляет

$$C = 7488 + 548 = 8036 \text{ рублей}$$

Экономия составляет

$$\Delta C = 8036 - 2152 = 5884 \text{ рублей}$$

Все расчеты ввелись в соответствии с критериями организации, в которой осуществляется процесс внедрения системы анкетирования, а именно:

- в филиале обучается около 700 студентов;
- анкетирование проводится один раз в год.

Таким образом, годовая экономия от внедрения информационной системы для сбора и анализа данных анкетирования составляет 5884 рублей

При расчетах окупаемости информационной системы анкетирования необходимо учитывать реальные затраты организации филиала ЮУрГУ на разработку данной системы. Так при расчетах реальных затрат, заработная плата программиста не учитывается.

Расчет затрат представлен в таблице 3.9. При расчетах использовались данные из таблицы 3.4

Таблица 3.9 – Расчет реальных затрат организации

п/п	Элементы затрат	Сумма, руб.
1	Расходы на приобретение, содержание и эксплуатацию программного обеспечения	1477,29
2	Обучение персонала	1440

Продолжение таблицы 3.9

п/п	Элементы затрат	Сумма, руб.
	Ввод в эксплуатацию	720
	Документирование	1980
	Оплата работы руководителя и консультантов	2000
	Итого	7617,29

Срок окупаемости рассчитывается по формуле (5).

$$\text{Ток} = K/\Delta C, \quad (5)$$

где Ток – срок окупаемости, лет;

К – полная стоимость владения системой, руб;

ΔC – годовая экономия.

В данном случае он равен:

$$\text{Ток } 7617,29/5884 = 1,29 .$$

Срок окупаемости = 1,29 лет.

Экономический эффект рассчитывается по формуле(6)

$$\text{Э} = \text{Эр} - \text{Ен} * K, \quad (6)$$

где Эр – годовая экономия, руб;

Ен – нормативный коэффициент ($\text{Ен}=0.15$).

В данном случае экономический эффект равен:

$$\text{Э} = 5884 - 0,15*7617,29=4741,41 \text{ рубль.}$$

Выводы по разделу три:

В данном разделе был проведен анализ расходов на приобретение, содержание и эксплуатацию программного обеспечения. Так же были произведены расчеты необходимых затрат на разработку системы и оценку экономической эффективности.

4 БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ

4.1 Требования к помещениям при работе за компьютером

В данном разделе используются нормы СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

В помещениях, предназначенных для эксплуатации ПЭВМ, обязательно должны иметь естественное и искусственное освещение. Расположение рабочих мест с ПЭВМ в подвальных помещениях запрещено.

Площадь рабочего места с компьютером и вспомогательным оборудованием должна составлять не менее 6м². Без вспомогательных устройств допускается площадь 4м² на одно рабочее место.

Помещения с ПЭВМ должны обязательно быть оборудованы заземлением. Также запрещается ставить ПЭВМ вблизи силовых кабелей, высоковольтных трансформаторов, технологического оборудования, которые могут создают помехи при работе с ПЭВМ.

4.2 Требования к освещению помещений и рабочих мест

В данном разделе используются нормы СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

В помещениях с ПЭВМ должно обязательно как естественное, так и искусственное освещение. Световой поток из оконного проема должен падать на рабочее место с левой стороны.

Искусственное освещение в помещениях с эксплуатацией ПЭВМ обязательно должен осуществляться системой общего равномерного освещения.

Освещенность поверхности стола должна быть от 300лк до 500лк. Разрешается установка светильников местного освещения для подсветки поверхности стола при работе с документами, но запрещено чтобы данное освещение создавала блики на поверхности экрана и увеличивала его освещенность более 300лк. Яркость светящихся поверхностей (окна, светильники), находящихся в поле зрения, должна быть не более 200 кд/м².

Отраженная блескость на рабочих поверхностях ограничивается за счет правильного выбора светильника и расположения рабочих мест по отношению к естественному источнику света. Яркость бликов на экране монитора не должна превышать 40 кд/м². Показатель ослепленности для источников общего искусственного освещения в помещениях должен быть не более 20, показатель дискомфорта в административно-общественных помещениях не более 40. Соотношение яркости между рабочими поверхностями не должно превышать 3:1 – 5:1, а между рабочими поверхностями и поверхностями стен и оборудования 10:1. Люминесцентные лампы, при применении общего освещения должны быть распложены в боковой стороне от рабочего места, параллельно уровню глаз пользователя в случае, когда экраны распложены в ряд.

В случае, когда компьютеры равномерно распделены по площади помещения, линии источников света должны быть распложены над рабочим местом ближе к передней части и направлены на оператора.

В целях обеспечения соответствующих параметров освещенности требуется проводить очистку осветительных приборов, а также окон и оконных поверхностей не менее чем два раза в год и вовремя осуществлять замену перегоревших ламп.

4.3 Требования к микроклимату

На рабочих местах пользователей персональных компьютеров должны обеспечиваться оптимальные параметры микроклимата в соответствии с

СанПиН 2.2.4.548-96. Согласно этому документу для категории тяжести работ 1а температура воздуха должна быть в холодный период года не более 22–24°C, в теплый период года 20-25°C. Относительная влажность должна составлять 40–60%, скорость движения воздуха – 0,1 м/с. Для поддержания оптимальных значений микроклимата используется система отопления и кондиционирования воздуха. Для повышения влажности воздуха в помещении следует применять увлажнители воздуха с дистиллированной или кипяченой питьевой водой.

Ионный состав воздуха должен содержать следующее количество отрицательных и положительных аэроионов; минимально необходимый уровень 600 и 400 ионов в 1 см³ воздуха; оптимальный уровень 3 000-5 000 и 1 500-3 000 ионов в 1 см³ воздуха; максимально допустимый – 50 000 ионов в 1 см³ воздуха.

4.4 Требования к шуму и вибрации в помещениях

Длительное пребывание персонала в рабочем помещении с высоким уровнем шума отрицательно сказывается на состоянии здоровья. Со временем человек начинает испытывать утомленность, головную боль, повышенное давление и общее недомогание. Все это приводит к значительному ухудшению производительности и качества работы. При длительном пребывании человека в помещении с интенсивностью шума свыше 80 дБа может привести к частичной или полной потере слуха.

В таблице 4.1 приводятся допустимые и эквивалентные параметры уровня звука по категориям тяжести и напряженности труда для различных видов трудовой деятельности и рабочих мест.

Таблица 4.1 – Предельно допустимые и эквивалентные уровни звука на рабочем месте для разных категорий труда в дБа

Трудовой процесс по категории напряженности	Категория тяжести процесса труда				
	Легкая нагрузка	Средняя нагрузка	Тяжелый труд		
			I степень	II степень	III степень
Легкая степень напряженности	80	80	75	75	75
Средняя степень напряженности	70	70	65	65	65
Напряженность труда I степени	60	60	-	-	-
Напряженность труда II степени	50	50	-	-	-

Уровни шума на рабочих местах пользователей персональных компьютеров не должны превышать значений, установленных СанПиН 2.2.4/2.1.8.562-96 и составляют не более 50 дБА. На рабочих местах в помещениях для размещения шумных агрегатов уровень шума не должен превышать 75 дБА, а уровень вибрации в помещениях допустимых значений по СН 2.2.4/2.1.8.566-96 категория 3, тип «в».

Снизить уровень шума в помещениях можно использованием звукопоглощающих материалов с максимальными коэффициентами звукопоглощения в области частот 63-8000 Гц для отделки стен и потолка помещений. Дополнительный звукопоглощающий эффект создают однотонные занавески из плотной ткани, повешенные в складку на расстоянии 15-20 см от ограждения. Ширина занавески должна быть в 2 раза больше ширины окна.

4.5 Требования к рабочему месту

В данном разделе используются нормы СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

Рабочие места с ПЭВМ должны располагаться так, чтобы естественный свет падал сбоку, желательно с левой стороны.

Обязательно нужно учитывать расстояние между рабочими столами с мониторами. Расстояние между боковыми поверхностями мониторов должен составлять не менее 1,2 м. Расстояние же между экраном и задней частью другого монитора не менее 2,0 м.

При выполнении творческой работы, которая требует высоких энергозатрат и умственного напряжения, рабочие места с ПЭВМ следует разделить друг от друга перегородками высотой 1,5 – 2,0 м.

Рабочий стол может быть любой конструкции, отвечающий современным требованиям эргономики. При этом конструкция должна быть такой, чтобы оборудование, применяемое на рабочей поверхности, было размещено оптимальным образом. Поверхность рабочего стола должна иметь коэффициент отражения от 0,5 до 0,7.

Конструкция рабочего кресла должна обеспечивать комфортное положение при работе с ПЭВМ, а также позволять изменить положения тела.

Рабочее кресло должно обязательно регулироваться по высоте и углам наклона сидения, спинки и расстоянию спинки от переднего края сиденья, быть подъемно-поворотным.

Эргономические требования к организации условий труда пользователя ЭВМ регламентируются следующими нормативными документами:

– ГОСТ 12.2.032-78. Рабочее место при выполнении работ сидя. Общие эргономические требования.

– СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы».

Обеспечения наилучших условий труда необходимо так, как повышает работоспособность и снижает утомляемость работников. Для этого следует организовать рабочее место согласно следующим рекомендациям:

– Высота рабочей поверхности для пользователей ПЭВМ должна регулироваться в пределах 680-800мм. При отсутствии регулировки высота рабочей поверхности стола должна составлять 725мм.

– Модульными размерами рабочей поверхности стола для ПЭВМ, на основании которых должны рассчитываться конструктивные размеры, следует считать ширину 800, 1000, 1200, 1400 мм, глубину 800 и 1000 мм при нерегулируемой его высоте, равной 725 мм.

– Также необходимо чтобы рабочий стол имел пространство для ног. Высота, которого должна составлять не менее 600 мм, а ширина не менее 500 мм. Глубина же на уровне колен – не менее 450 мм и на уровне вытянутых ног – не менее 650 мм.

Также немало важным является конструкция рабочего кресла.

Конструкцией рабочего кресла должны быть обеспечены:

– ширина и глубина поверхности сиденья не меньше 400 мм;
– поверхность сиденья, которая имеет закругленный передний край;
– регулировка по высоте сиденья в пределах 400 – 550 мм и углом наклона вперед до 15 град. и назад до 5 град.;

– высота опорной поверхности спинки 300 ± 20 мм, ширина - не меньше 380 мм и радиус кривизны горизонтальной плоскости 400 мм;

– угол наклона спинки в вертикальной плоскости в пределах 0 ± 30 град.;

– регулировка расстояния спинки от переднего края сиденья от 260 до 400 мм;

– стационарные или съемные подлокотники длиной не менее 250 мм и шириной от 50 до 70 мм;

– регулировка подлокотников по высоте над сиденьем в пределах 230±30 мм и внутреннего расстояния между подлокотниками от 350 до 500 мм.

Располагать клавиатуру на рабочей плоскости стола необходимо на расстоянии 100 – 300 мм от переднего края, который обращен к пользователю или на специальной рабочей поверхности, регулируемой по высоте и изолированной от основной столешницы.

4.6 Режим труда и отдыха при работе за компьютером

Режим труда и отдыха предусматривает соблюдение определенной длительности непрерывной работы на ПК и перерывов, регламентированных с учетом продолжительности рабочей смены, видов и категории трудовой деятельности.

СанПиН 2.2.2/2.4.1340-03 предусматривает разделение трудовой деятельности при работе с ПЭВМ на три группы: группа А – работа по считыванию информации с экрана с предварительным запросом; группа Б - работа по вводу информации; группа В – творческая работа в режиме диалога с ПК.

В случае, когда в течение рабочей смены пользователь при взаимодействии с ПЭВМ выполняет разные виды работ, основной работой считается та, на которую отводится не менее 50% рабочего времени.

По видам трудовой деятельности разделяют три категории тяжести и напряженности при работе с ПЭВМ, уровень нагрузки для которых определяется как: суммарное число считываемых знаков – группа А; суммарное число считываемых или вводимых знаков – группа Б; общее время проведенной работы на ПЭВМ – группа В.

В таблице 4.2 приводятся категории тяжести и напряженности в зависимости от вида работы с ПЭВМ.

Таблица 4.2 – Категории тяжести и напряженности в зависимости от вида работы с ПЭВМ

Категории тяжести работы с ПЭВМ	Нагрузка при видах работ за рабочую смену с ПЭВМ			Регламентированные перерывы, мин	
	группа А, кол-во знаков	группа Б, кол-во знаков	группа В, часов	Смена 8 часов	Смена 12 часов
I категория	До 20 тыс	До 15 тыс	2,0	50	80
II категория	До 40 тыс	До 30 тыс	4,0	70	110
III категория	До 60 тыс	До 40 тыс	6,0	90	140

Суммарное время регламентированных перерывов устанавливается в зависимости от категории трудовой деятельности и уровня нагрузки за рабочую смену.

Рекомендуется организовывать рабочий день таким образом, чтобы работа с использованием персонального компьютера чередовалась с какой-либо другой, в которой не используется ПЭВМ. Это необходимо для предотвращения преждевременной утомляемости работника.

При работе, характер которой предполагает постоянное взаимодействие с ВДТ (ввод текстов или набор данных и т.п.) с высокой степенью сосредоточенности и внимания, в случае, когда переключение на другие виды трудовой деятельности, не связанные с ПЭВМ невозможно, необходима организация перерывов на 10 – 15 мин через каждые 45 - 60 мин. работы.

При непрерывной работе с видеотерминалами без регламентированного перерыва продолжительности не должна превышать 1 ч.

Независимо от категории и вида трудовой деятельности при работе с ПЭВМ в ночную смену следует увеличить продолжительность регламентированных перерывов на 30%.

В регламентированные перерывы, которые предназначены для снижения нервного и эмоционального напряжения, утомления и устранения влияния гиподинамии и гипокинезии, следует выполнять комплекс упражнений для глаз, с плечевого пояса и рук, туловища и ног. Целесообразно менять комплексы упражнений через 2 –3 недели.

Регламентированные перерывы и микропаузы целесообразно использовать для выполнения комплекса упражнений и гимнастики для глаз, пальцев рук, а также массажа. Комплексы упражнений целесообразно менять через 2 – 3 недели.

Для пользователей с высоким уровнем напряженности при работе с персональным компьютером, рекомендуется отдых в специально оборудованных помещениях.

4.7 Медико-профилактические и оздоровительные мероприятия

Согласно СанПиН 2.2.2/2.4.1340-03, все профессиональные пользователи ПК должны проходить обязательные предварительные медицинские осмотры при поступлении на работу, периодические медицинские осмотры с обязательным участием терапевта, невропатолога и окулиста, а также проведением общего анализа крови и ЭКГ.

Женщины со времени установления беременности переводятся на работы, не связанные с использованием ПЭВМ, или для них ограничивается время работы с ПЭВМ (не более 3-х часов за рабочую смену) при условии соблюдения гигиенических требований.

Близорукость, дальнозоркость и другие нарушения рефракции должны быть полностью скорректированы очками. Для работы должны использоваться очки, подобранные с учетом рабочего расстояния от глаз до экрана дисплея. При более серьезных нарушениях состояния зрения вопрос о возможности работы на ПК решается врачом-офтальмологом.

Досуг рекомендуется использовать для пассивного и активного отдыха (занятия на тренажерах, плавание, езда на велосипеде, бег, игра в теннис, футбол, лыжи, аэробика, прогулки и т.п.). Дважды в год (весной и поздней осенью) рекомендуется проводить курс витаминотерапии в течение месяца. Категорически должно быть запрещено курение на рабочих местах и в помещениях с ПК.

4.8 Требования к электробезопасности

Нормы на допустимые токи и напряжения прикосновения в электроустановках должны устанавливаться в соответствии с предельно допустимыми уровнями воздействия на человека токов и напряжений прикосновения и утверждаться в установленном порядке согласно ГОСТ 12.1.038-82 ССБТ «Электробезопасность. Предельно допустимые значения напряжений прикосновения и токов».

Очень важно уделять большое внимание безопасности при работе с ПЭВМ на рабочем месте. Не рекомендуется очищать ПЭВМ от пыли во включенном состоянии и работать непосредственно с компьютером во влажной одежде и влажными руками.

Перед началом работы с персональным компьютером необходимо убедиться в отсутствии различных свисающих проводов со стола, в целостности состояния проводом электропитания, в отсутствии повреждений рабочей аппаратуры, а также убедиться в наличии заземления экранного фильтра.

В процессе работы токи статического электричества, наведенные на корпус монитора, системного блока и клавиатуры, могут приводить к разрядам при прикосновении к этим элементам. Опасности для пользователя это не представляет, но может привести к перебоям при работе оборудованием. Снизить уровень токов статического электричества можно, используя нейтрализаторы, местное и общее увлажнители воздуха, покрытие полов с антистатической пропиткой.

Выводы по разделу четыре:

В данном разделе мы провели анализ опасных и вредных производственных факторов изучили требования к безопасности и на основании их разработали защитные мероприятия и инструкцию по охране труда.

Решение проблемы безопасности жизнедеятельности состоит в обеспечении нормальных (комфортных) условий деятельности людей, в защите человека и окружающей его среды от воздействия вредных факторов, превышающих нормативно-допустимые уровни. Поддержание оптимальных условий деятельности и отдыха человека создает предпосылки для высокой работоспособности и продуктивности

В ходе осмотра рабочих мест в организации не было выявлено нарушений правил безопасности жизнедеятельности.

ЗАКЛЮЧЕНИЕ

В результате проделанной работы была разработана информационная система для сбора и обработки данных анкетирования. Данная система была успешно внедрена в филиале ЮУрГУ. Помимо этого было приведено экономическое обоснование внедрения данной информационной системы.

Преследуемая изначально цель автоматизации процесса анкетирования и обработки данных достигнута.

Разработанная система позволит повысить скорость обработки информации, сократит сроки формирования отчетов и сэкономит время работы пользователей.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1 «О государственной регистрации юридических лиц и индивидуальных предпринимателей»: Федеральный закон от 08.08.2001 N 129-ФЗ: (ред. От 30.03.2015, с изм. От 18.05.2015) // «Собрание законодательства РФ», 13.08.2001, N 33 (часть I), ст. 3431

2 «О защите прав юридических лиц и индивидуальных предпринимателей при осуществлении государственного контроля (надзора) и муниципального контроля»: Федеральный закон от 26.12.2008 N 294-ФЗ: (ред. От 06.04.2015) // «Российская газета», № 4823 от 30.12.2008 г.

3 Налоговый кодекс Российской Федерации от 05.08.2000 № 117-ФЗ // «Собрание законодательства РФ», 07.08.2000, N 32, ст. 3340.

4 ГОСТ 12.1.005-88 Система стандартов безопасности труда. Общие санитарно-гигиенические требования к воздуху рабочей зоны – М.: ИПК Издательство стандартов, 2002

5 ГОСТ 12.0.003-74 Система стандартов безопасности труда. Опасные и вредные производственные факторы. Классификация – М.: ИПК Издательство стандартов, 2002.

6 ГОСТ 12.1.038-82 Система стандартов безопасности труда. Электробезопасность. Предельно допустимые значения напряжений прикосновения и токов – М.: Издательство стандартов, 1988.

7 СП 52.13330.2011 «Свод правил. Естественное и искусственное освещение. Актуализированная редакция СНИП 23-05-95» - М.: Минрегион России, 2011.

8 СанПиН 2.2.2/2.4.1340-03 «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы». – М.: Минздрав России, 2003.

9 СТО ЮУрГУ 22-2008 Стандарт организации. Основные положения подготовки, проведения и оценки защиты выпускной квалификационной работы

(проекта) студента / сост. Т.И. Парубочая, Н.В. Сырейщикова, С.Д. Ваулин, В.И. Гузеев, Б.А. Кулаков. – Челябинск: Изд-во ЮУрГУ, 2008. – 34 с.

10 Рихтер, Дж. CLR via C#. Программирование на платформе Microsoft .NET Framework 4.0 на языке C#. / Дж. Рихтер – 3-е изд. – СПб.: Питер, 2012.- 928 с.

11 Стиллмен, Э. Изучаем C#. / Э. Стиллмен, Дж. Грин. – 3-е изд. – СПб.: Питер, 2014. – 816 с.

12 Подбельский, В.В. Язык C#. Базовый курс. / В.В. Подбельский – 2-е изд. – Москва: Инфра-М, 2013. – 426 с.

13 Тепляков, С.В. Паттерны проектирования на платформе .NET. / С.В. Тепляков – СПб.: Питер, 2015. — 320 с.

14 Ржеуцкая, С.Ю. Базы данных. Язык SQL: учеб.пособие / С.Ю. Ржеуцкая – Вологда: ВоГТУ, 2010. – 159с.

15 Беляков, Г.И. Безопасность жизнедеятельности. Охрана труда: Учебник для бакалавров / Г.И. Беляков. – М.: Юрайт, 2013. – 572 с.

16 Михайлова, Э.А. Экономическая оценка инвестиций: учебное пособие. / Э.А. Михайлова, Л.Н. Орлова – Рыбинск: РГАТА, 2008. – 176 с.

17 Руководство по Entity Framework. – <https://metanit.com/sharp/entityframework/>

18 Общие сведения о платформе .NET Framework – [https://msdn.microsoft.com/ru-ru/library/zw4w595w\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/zw4w595w(v=vs.110).aspx)

19 Работа с базами данных в .NET Framework – https://professorweb.ru/my/ADO_NET/base/level1/info_db.php

20 Введение в запросы LINQ (C#) – [https://msdn.microsoft.com/ru-ru/library/bb397906\(v=vs.110\).aspx](https://msdn.microsoft.com/ru-ru/library/bb397906(v=vs.110).aspx)

ПРИЛОЖЕНИЯ

ПРИЛОЖЕНИЕ А. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Руководство пользователя программы
Анкетирование: Преподаватель глазами студента

1. НАЗНАЧЕНИЕ СИСТЕМЫ

Система «Анкета Преподаватель глазами студента» предназначена для проведения анкетирования среди студентов филиала.

Основной целью данной системы является автоматизация процесса анкетирования, а так же автоматизированная обработка результатов анкетирования и составление отчетов.

2. УСЛОВИЯ ПРИМЕНЕНИЯ СИСТЕМЫ

Квалификация пользователя:

уверенный пользователь ПК

3. ПОДГОТОВКА СИСТЕМЫ К РАБОТЕ

1 Настройка сервера

Для работы приложения необходимо иметь подключение к серверу баз данных MySQL.

2 Настройка подключения

Запустите приложение «Анкета Преподаватель глазами студента». В отрывшемся окне отметьте опцию Настройка подключения.

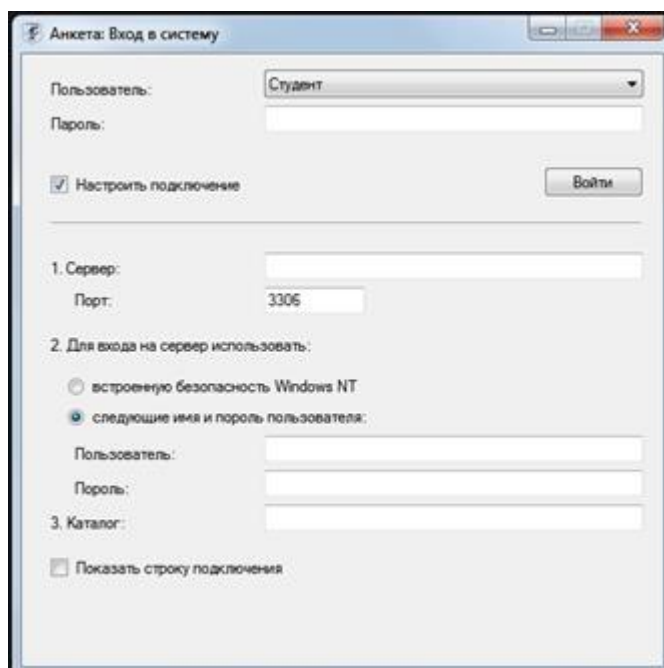


Рисунок А.1 – Настройка подключения

Для осуществления настройки подключения заполните поля:

- Сервер – укажите адрес сервера, на котором находится база данных;
- Порт – укажите порт подключения к серверу (3306 – стандартный порт подключения);
- Для входа на сервер рекомендуется использовать имя и пароль пользователя базы данных;
- Каталог – название базы данных.

В некоторых случаях может потребоваться более тонкая настройка подключения. Для этого выберите пункт Показать строку подключения. Отобразится строка подключения к базе данных, которую можно отредактировать вручную, прописав дополнительные параметры подключения.

Рисунок А.2 – Пример настройки подключения.

Для того чтобы приложение автоматически при каждом запуске осуществляло подключение к базе данных по заданному адресу, скопируйте строку подключения из одноименного поля приложения, затем в конфигурационном файле «Анкета Преподаватель глазами студента.exe.config» найдите секцию `<connectionStrings>` и замените значение параметра `connectionString` новой строкой подключения.

```

</configSections>
<connectionStrings>
  <add name="UniverContext" connectionString="Server=127.0.0.1; Port=3306; Uid=newuser; Pwd=12345;
  Database=UniverDB3;" providerName="MySql.Data.MySqlClient" />
</connectionStrings>
<startup>
  <supportedRuntime version="v4.0" sku=".NETFramework,version=v4.5.2" />
</startup>

```

Рисунок А.3 – Конфигурационный файл.

4. ОПИСАНИЕ ОПЕРАЦИЙ

1 Первый запуск системы

Прежде чем начать работу с системой Анкетирование необходимо осуществить первый запуск.

Запустите приложение «Анкета Преподаватель глазами студента». В открывшемся окне Анкета: Вход в систему в графе Пользователь выберите Администратор, введите пароль и нажмите кнопку Войти.

В случае успешного подключения к серверу и заполнения системы данными, на экране появится следующее сообщение.

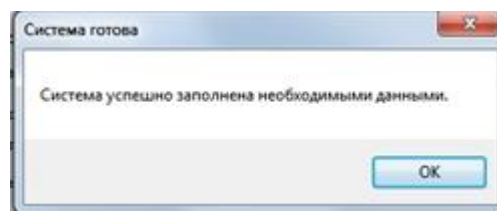


Рисунок А.4 – Сообщение о готовности системы

2 Подготовка к анкетированию

Прежде чем провести анкетирование в группе, составьте список преподавателей, которых необходимо указать в анкете. В соответствии с этим списком студенты будут добавлять фамилии преподавателей в анкеты (Данный процесс описан в разделе 3.2 Редактирование списка преподавателей).

3 Анкетирование студентов

3.1 Запуск приложения

Чтобы пройти анкетирование, необходимо запустить приложение «Анкета Преподаватель глазами студента».

В открывшемся окне Анкета: Вход в систему в графе Пользователь выберите Студент и нажмите кнопку Войти.

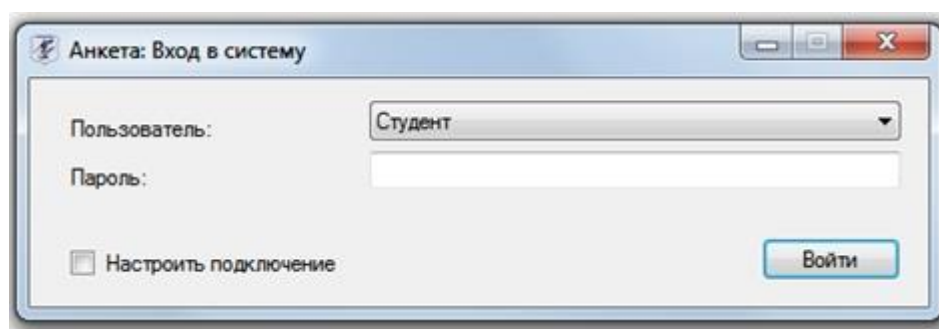


Рисунок А.5 – Окно Анкета: Вход в систему

Перед вами откроется окно Анкета: Студент, на котором находится анкета. Прежде чем заполнять ее, необходимо добавить в анкету преподавателей.

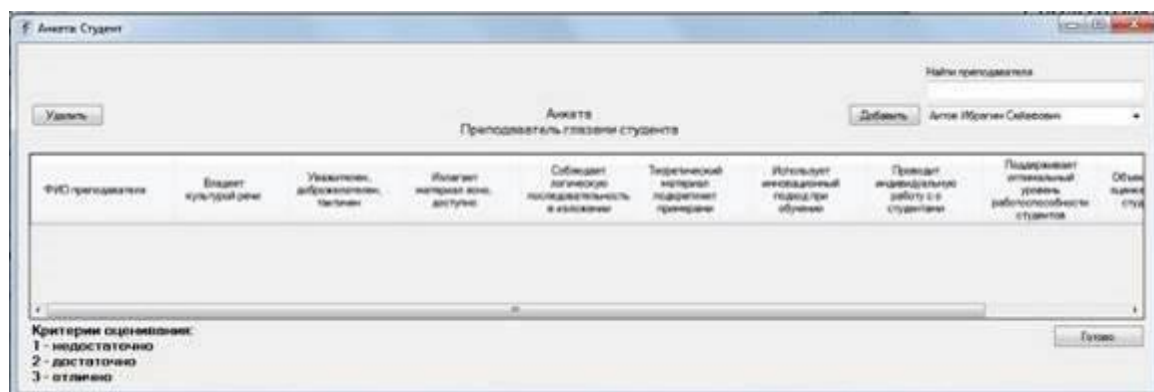


Рисунок А.6 – Окно Анкета: Студент

3.2 Редактирование списка преподавателей

Для этого можно воспользоваться строкой поиска либо найти преподавателя в выпадающем списке. Нажмите кнопку Добавить, чтобы добавить фамилию выбранного преподавателя в анкету.

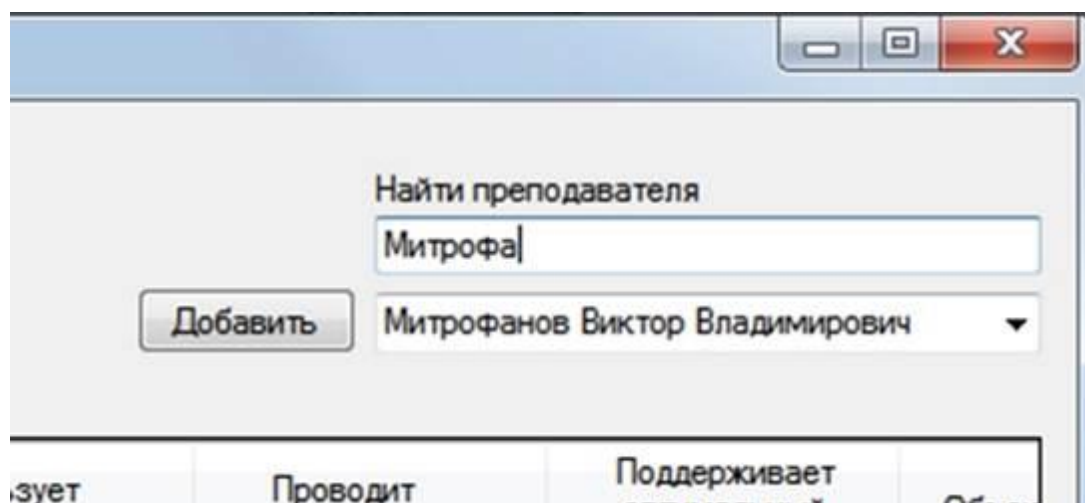


Рисунок А.7 – Добавление преподавателя в анкету

Таким же образом добавьте фамилии других преподавателей.

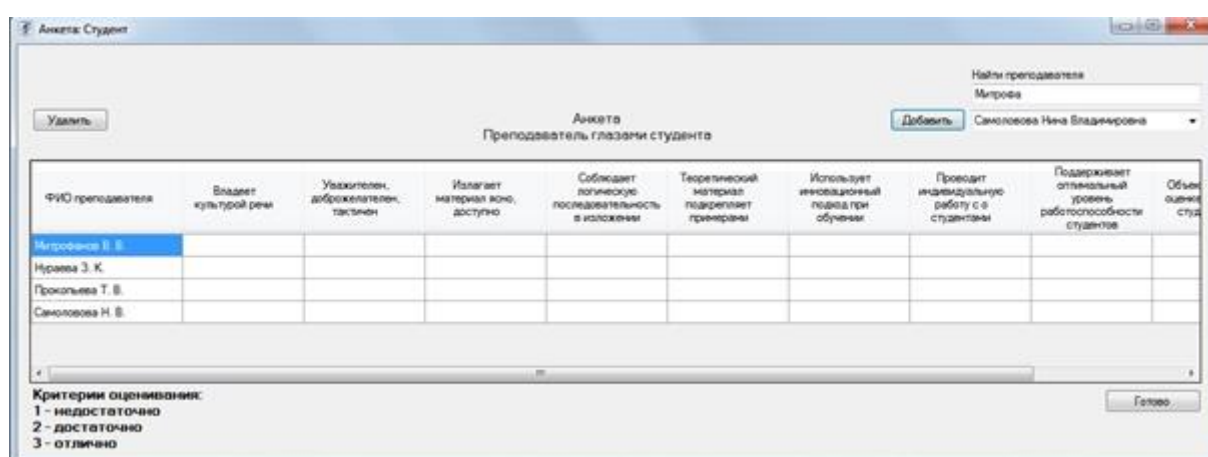


Рисунок А.8 – Анкета готова к заполнению

Если вы случайно добавили в анкету не ту фамилию, выделите ее в анкете и нажмите кнопку «Удалить».

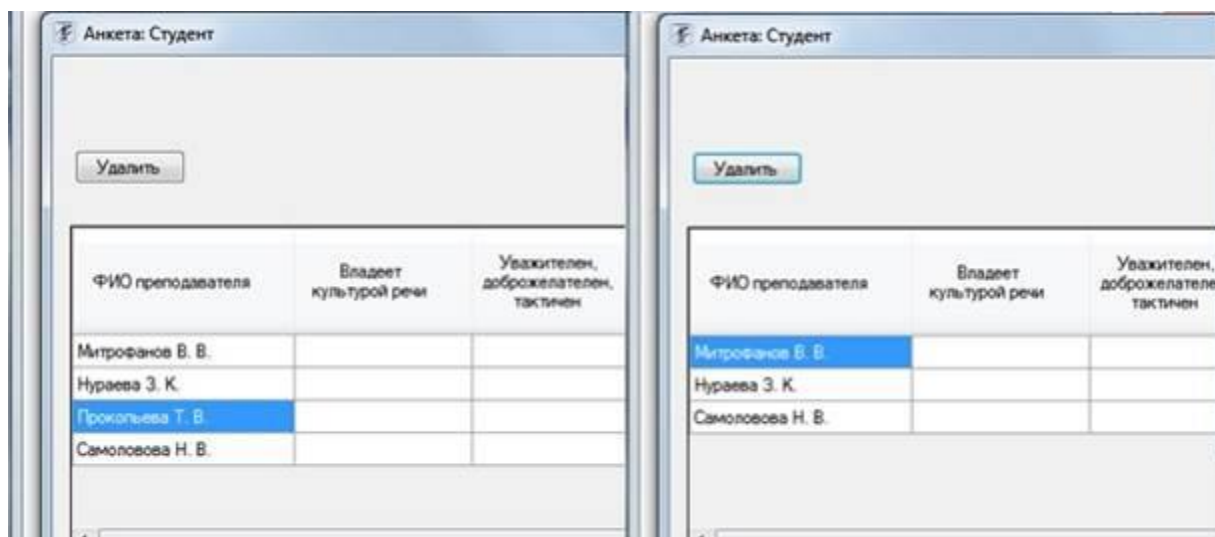


Рисунок А.9 – Удаление преподавателя из анкеты

Добавлять и удалять фамилии преподавателей можно в любой момент времени заполнения анкеты.

3.3 Заполнение анкеты

При заполнении анкеты обратите внимание на Критерии оценивания. Все поля (кроме последней колонки) обязательны к заполнению и заполняются в соответствии с этими критериями. В последней колонке Особое мнение можете указать свое мнение о преподавателе, оставить комментарий или свои пожелания. Данная колонка не обязательна к заполнению, но ваше мнение будет учтено.

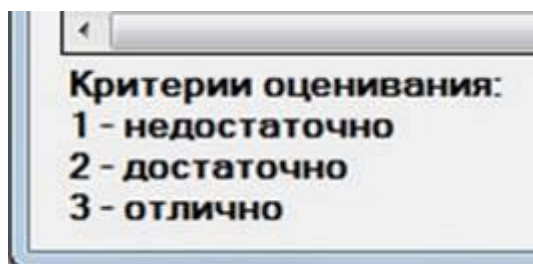


Рисунок А.10 – Критерии оценивания

Прежде чем завершить заполнение анкеты, убедитесь, что вы указали в анкете всех преподавателей и заполнили все необходимые поля.

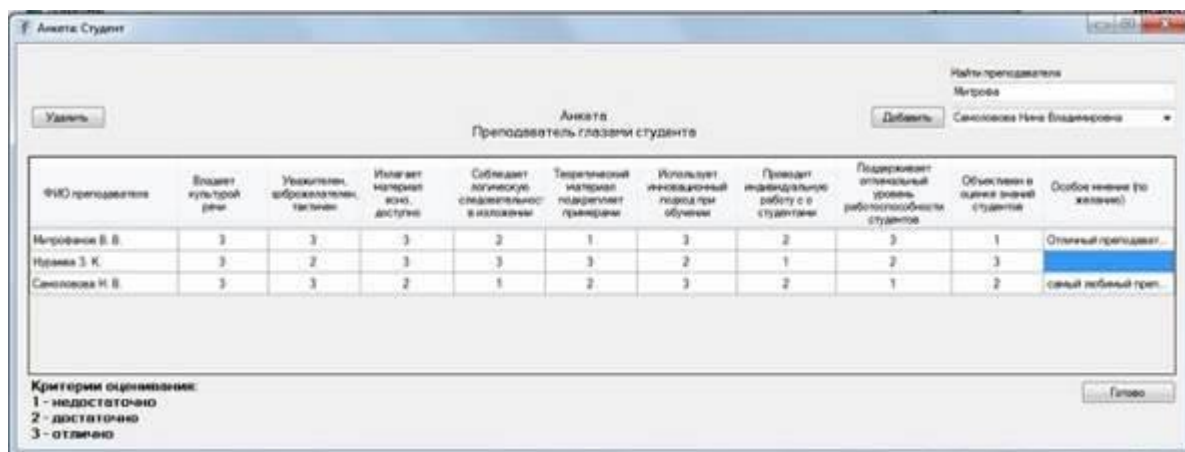


Рисунок А.11 – Заполненная анкета

3.4 Завершение анкетирования

Когда вы будете готовы завершить анкетирование, нажмите кнопку «Готово».

В случае правильного заполнения анкеты вы получите соответствующее сообщение. Нажмите кнопку «Ок», программа сама завершит свою работу.

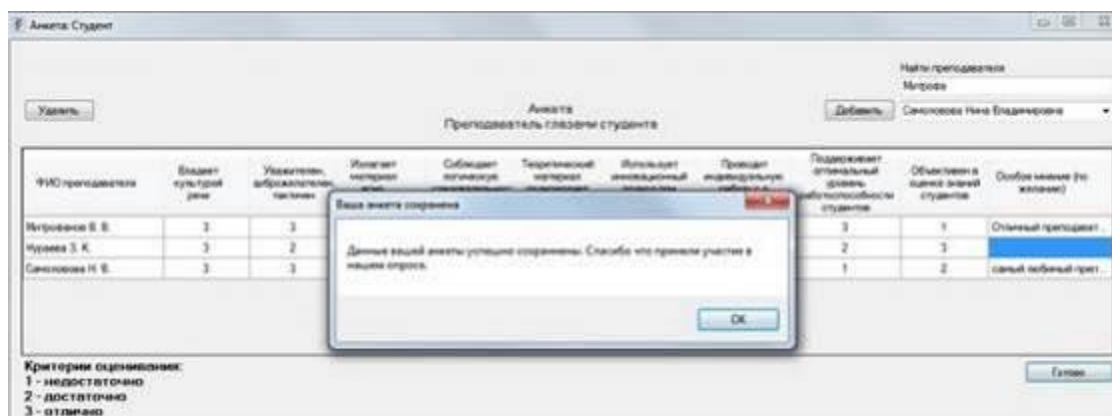


Рисунок А.12 – Завершение анкетирования

Если вы заполнили анкету неверно, вы получите соответствующее сообщение. Нажмите кнопку «Ок» и исправьте это.

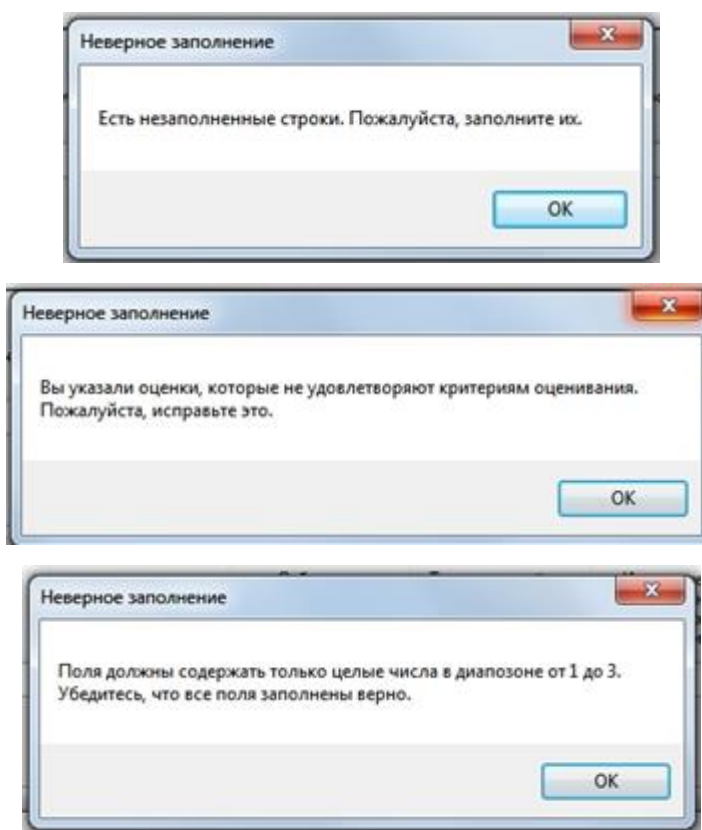


Рисунок А.13 – Неверное заполнение анкеты

4 Обработка результатов анкетирования

Для того, что бы посмотреть отчеты проведенного анкетирования необходимо запустить приложение «Анкета Преподаватель глазами студента».

В открывшемся окне «Анкета: Вход в систему» в графе Пользователь выберите «Администратор», введите пароль и нажмите кнопку «Войти».

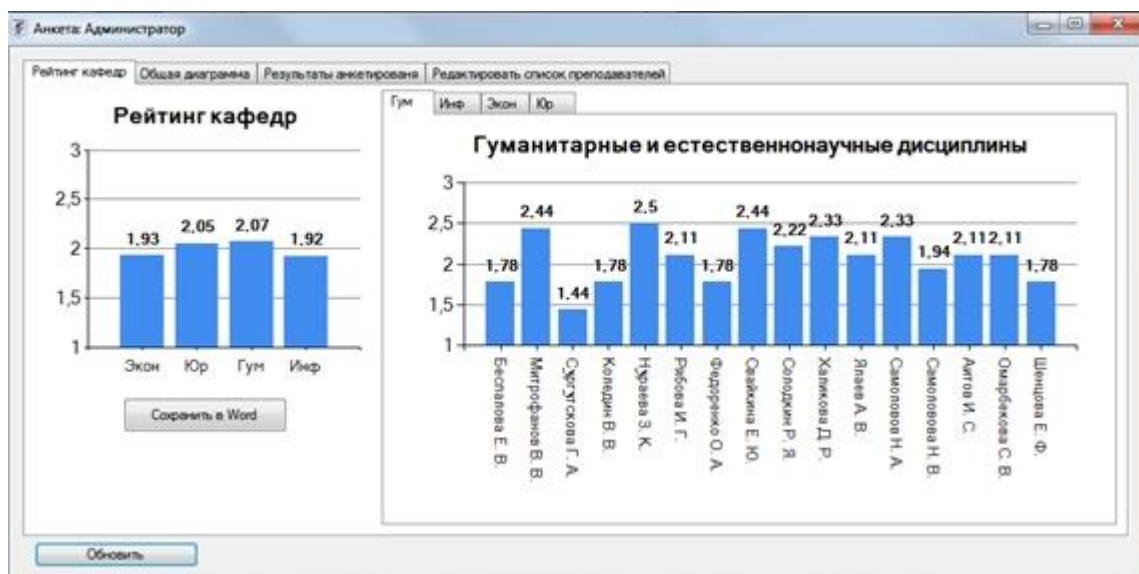


Рисунок А.14 – Окно Анкета: Администратор

4.1 Тестовые данные

Работоспособность системы можно проверить на тестовых данных.

Для этого перейдите на вкладку «Результаты анкетирования» и нажмите кнопку «Тесты».



Рисунок А.15 – Добавить тесты

После того, как программа заполнит систему тестовыми данными, нажмите кнопку «Обновить».

4.2 Удаление тестовых данных

После того, как вы проверите работоспособность системы на тестовых данных, рекомендуется сразу удалить их из системы, чтобы впоследствии они не попали в финальный отчет.

Для удаления из системы тестовых данных перейдите на вкладку Результаты анкетирования и нажмите кнопку «Удалить тесты».



Рисунок А.16 – Удалить тесты

4.3 Сохранение результатов анкетирования

Перед сохранением результатов анкетирования убедитесь, что из системы удалены тестовые данные (см. раздел 4.2 Удаление тестовых данных).

Для того, чтобы сохранить результаты анкетирования в формате Word перейдите на вкладку Рейтинг кафедр и нажмите кнопку «Сохранить в Word».

В появившемся окне выберите папку для сохранения результатов и нажмите кнопку «Ок». Система автоматически сохранит результаты анкетирования.

4.4 Редактирование отчета о результатах анкетирования

Для настройки отображаемых результатов перейдите на вкладку «Результаты анкетирования».

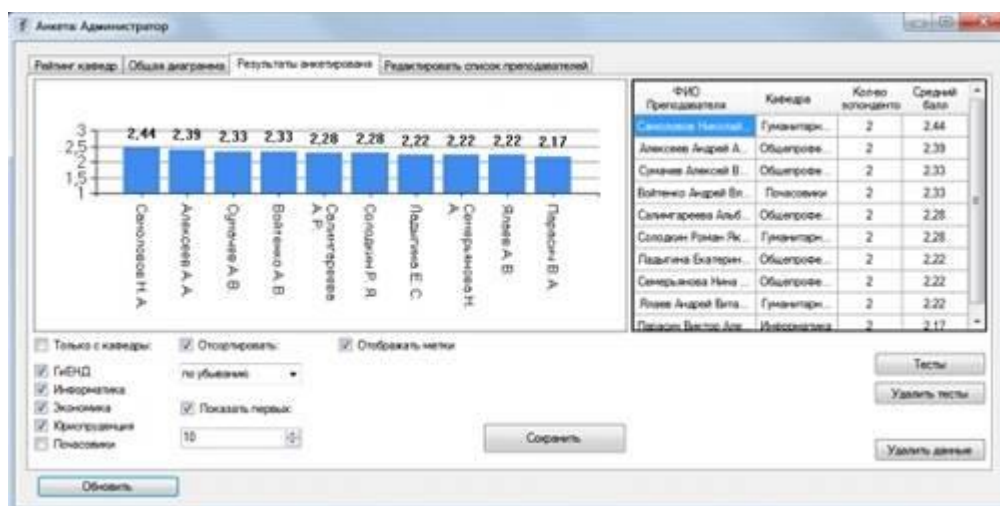


Рисунок А.17 – Вкладка Результаты анкетирования

На этой вкладке расположены фильтры для настройки отображаемых данных.

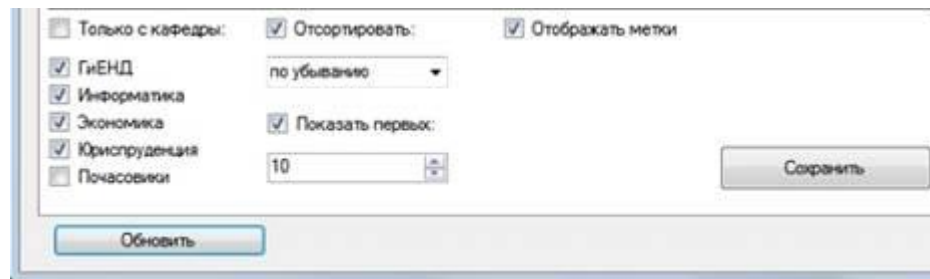


Рисунок А.18 – Фильтры

С помощью фильтров можно:

- установить сортировку по среднему баллу или по алфавиту;
- отображать преподавателей с определенных кафедр;
- настроить число отображаемых преподавателей;
- настроить отображение меток на диаграмме.

После настройки отображения результатов анкетирования, можно сохранить отчет с текущими настройками. Для этого нажмите кнопку «Сохранить» и в открывшемся диалоговом окне выберете имя файла для сохранения отчета.

5 Редактирование списка преподавателей

Для редактирования списка преподавателей перейдите на вкладку «Редактирование списка преподавателей».

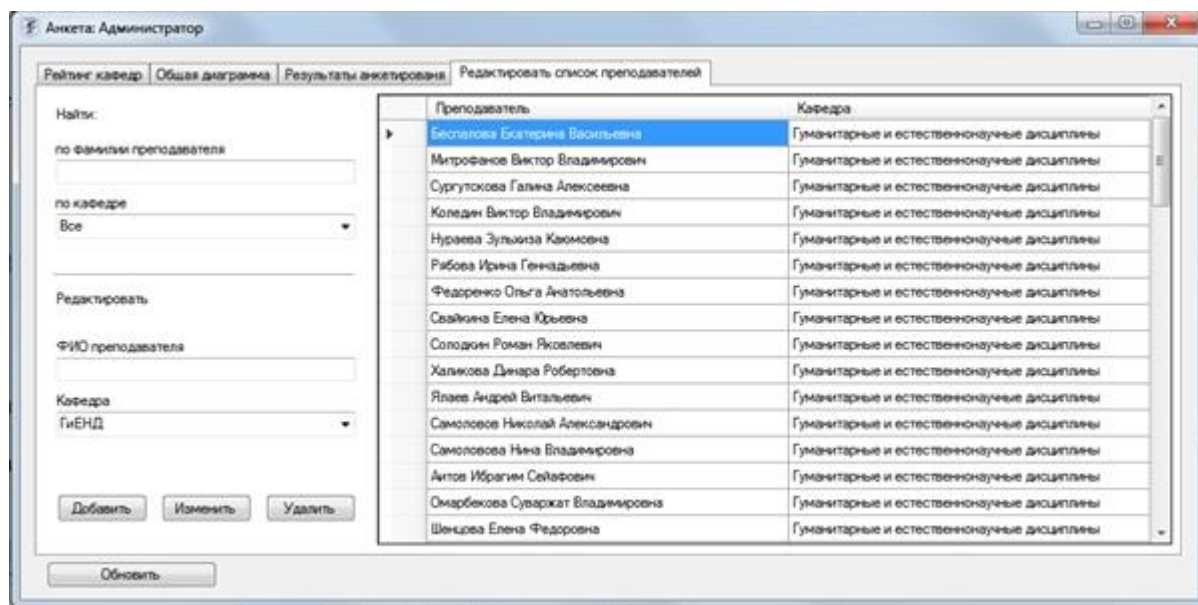


Рисунок А.19 – Вкладка Редактирование списка преподавателей

На этой вкладке можно добавлять в систему преподавателей, редактировать и удалять их.

Примечание: преподаватели, которые не закреплены ни за одной из кафедр, отмечаются как Почасовики.

Почасовик отображаются на общей диаграмме. Если вы не хотите, чтобы эти преподаватели отображались в общих результатах, воспользуйтесь редактированием отчета (см. раздел 4.4 Редактирование отчета о результатах анкетирования).

5.1 Добавление преподавателя

Чтобы добавить преподавателя в систему, в области «Редактировать» введите ФИО преподавателя, выберите кафедру и нажмите кнопку «Добавить».

Рисунок А.20 – Область редактирования

5.2 Редактировать информацию о преподавателе

Для того чтобы изменить имя или кафедру преподавателя, выберите его в таблице справа. Для облегчения процесса поиска преподавателя в области «Найти» можно воспользоваться поисковой строкой или выбрать кафедру из выпадающего списка.

Рисунок А.21 – Область Поиска

После того как вы выбрали преподавателя воспользуйтесь областью редактирования чтобы изменить имя преподавателя или кафедру. После этого нажмите кнопку «Изменить».

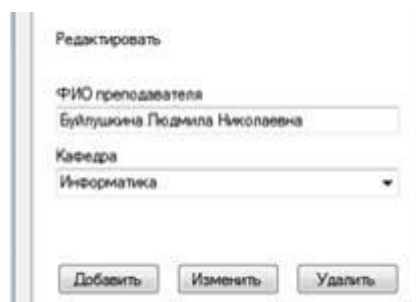


Рисунок А.22 – Область редактирования

5.3 Удалить информацию о преподавателе

Для того, чтобы удалить информацию о преподавателе из системы, выберите преподавателя из таблицы (можете воспользоваться областью поиска) и в области редактирования нажмите кнопку «Удалить».

6 Удаление результатов анкетирования

Результаты анкетирования можно удалить из системы, если больше нет нужды хранить их в базе данных.

Для этого перейдите на вкладку Результаты анкетирования и нажмите кнопку «Удалить данные».

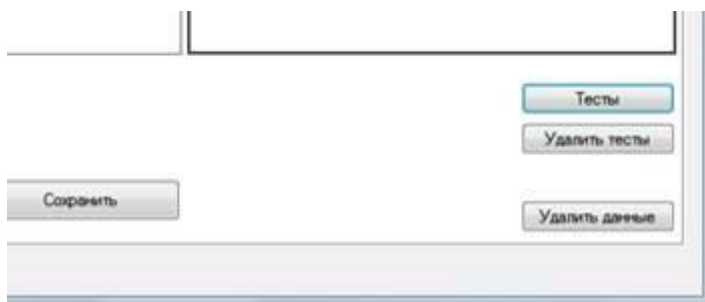


Рисунок А.23 – Область редактирования

Откроется окно подтверждения удаления данных. Для подтверждения удаления данных введите в поле пароль администратора и нажмите кнопку «Удалить».

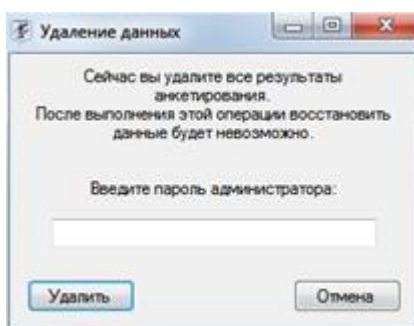


Рисунок А.24 – Область Поиска

5. АВАРИЙНЫЕ СИТУАЦИИ

При сбое в работе аппаратуры восстановление нормальной работы системы должно производиться после:

- перезагрузки операционной системы;
- запуска исполняемого файла системы.

При неверных действиях пользователей, неверных форматах или недопустимых значениях входных данных, система выдает пользователю соответствующие сообщения, после чего возвращается в рабочее состояние, предшествовавшее неверной (недопустимой) команде или некорректному вводу данных.

При возникновении других ошибок и неисправностей перезапустите систему. При повторном возникновении данных ошибок осуществите настройку подключения к серверу БД.

ПРИЛОЖЕНИЕ Б. КОМПАКТ-ДИСК

Содержание:

1. Пояснительная записка;
2. Презентация;
3. Приложение «Анкетирование Преподаватель глазами студента»;
4. Исходный код программы.
5. Руководство пользователя