

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Кафедра Информационные системы

«Допущен к защите»
Заведующий кафедрой _____

(Ф.И.О., ученая степень, звание)

« _____ » _____ 20 _____ г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Разработка и проектирование автоматизированной информационной системы "Учет деятельности преподавателей кафедры "Информационные системы"

Специальность 5B070300 - Информационные системы

Выполнил (а) Ситниченко Т.В. ИС 10-2
(Фамилия и инициалы) группа

Научный руководитель Кунярова А.П. ст. преподав.
(Фамилия и инициалы, ученая степень, звание)

Консультанты:

по экономической части:

Сейкина А.С. к.э.н., доцент
(Фамилия и инициалы, ученая степень, звание)
« 26 » _____ 20 14 г.
(подпись)

по безопасности жизнедеятельности:

Башкирева д.с. ст. преподав.
(Фамилия и инициалы, ученая степень, звание)
« 5 » _____ 20 14 г.
(подпись)

по применению вычислительной техники:

Касрабаева Б.К. м.т.н. ст. преподаватель
(Фамилия и инициалы, ученая степень, звание)
« 05 » _____ 20 14 г.
(подпись)

Нормоконтролер:

Али А.Т. к.т.н., доцент
(Фамилия и инициалы, ученая степень, звание)
« 09 » _____ 20 14 г.
(подпись)

Рецензент:

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

Алматы 2014 г.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Факультет "Информационные технологии"
Специальность 58070300 - "Информационные системы"
Кафедра "Информационные системы"

ЗАДАНИЕ

на выполнение дипломного проекта

Студент Ситниченко Тамара Владимировна
(фамилия, имя, отчество)

Тема проекта Разработка и проектирование автоматизи-
рованной информационной системы "Учет учетности
преподавателей кафедры "Информационные системы"
утверждена приказом ректора № от « » сентября 20 г.

Срок сдачи законченной работы « » 20 г.

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

Исходные данные задание к дипломному проекту
Требуемые параметры результатов проектирования:
модель информационной модели для расчета ритмики
преподавателей, ER-диаграмма база данных, програм-
мная реализация приложения, база данных инфор-
мационной системы

Исходные данные объекта: результаты исследования предмет-
ной области
Перечень подлежащих разработке дипломного проекта вопросов или краткое содержание дипломного проекта:

1. Анализ методики оценки качества работы преподава-
телей и заведующего кафедрой
2. Разработка и проектирование автоматизированной
информационной системы
3. Программная реализация разработанной АИС
4. Технич.-экономическое обоснование проекта
5. Анализ опасных и вредных производственных факто-
ров.

Перечень графического материала (с точным указанием обязательных чертежей)

1. Алгоритм разработки запросов пользователем
2. Диаграмма прецедентов
3. Диаграмма видов деятельности
4. Диаграмма последовательности
5. Диаграмма компонентов
6. Диаграмма классов
7. ER-диаграмма баз данных (логический уровень)

Рекомендуемая основная литература

1. Ч. Боггс, и Боггс 'UML и Rational Rose 2002' - Издательство "ЛОРЧ", 2004
2. Введение в UML лекции [Электронный ресурс] -
Путь доступа: <http://www.intelit.ru> - 30.01.2010
3. Бекмиева А. И. Методические указания к выполнению
английской части дипломной работы для бакалавров
специальности 5В070300 - Информационные системы.
Астана: АУЭС; 2013. - 24с.

Консультанты по проекту с указанием относящихся к ним разделов

Раздел	Консультант	Сроки	Подпись
Технич. часть	Бекмиева А. И.		А. И.
БД	Бекмиева А. И.		А. И.

ГРАФИК
подготовки дипломного проекта

№ п/п	Наименование разделов, перечень разрабатываемых вопросов	Сроки представления руководителю	Примечание
1	Описание предметной области. Анализ методики оценки качества работы преподавателей и заведующих кафедр	20.01.2014г - 21.02.2014г	
2	Разработка и проектирование АИС "Учет деятельности преподавателей кафедры "Информационные системы"	24.02.2014г - 21.04.2014г	
3	Проектирование базы данных программной реализации разработанного АИС. Проектирование пользовательского интерфейса	22.04.2014г - 12.05.2014г	
4	Технико-экономическое обоснование проекта	12.05.2014г - 26.05.2014г	
5	Безопасности жизнедеятельности. Анализ опасностей и вредных производственных факторов	20.04.2014г - 5.05.2014г	

Дата выдачи задания « » 20 г.

Заведующий кафедрой _____
(подпись) (Фамилия и инициалы)

Руководитель _____
(подпись) (Фамилия и инициалы)

Задание принял к исполнению студент _____
(подпись) (Фамилия и инициалы)

Аннотация

В дипломном проекте разработана автоматизированная информационная система «Учет деятельности преподавателей кафедры «Информационные системы». Осуществлен анализ существующей системы расчета рейтинга на кафедре Информационных систем, разработаны UML диаграммы и база данных АИС «Учет деятельности преподавателей кафедры «Информационные системы»».

АИС «Учет деятельности преподавателей кафедры «Информационные системы»» разработана с целью автоматизации расчета рейтинга преподавателей и заведующего кафедры Информационных систем, а также для создания единой базы данных кафедры.

Annotation

In the thesis project developed an automated information system "Accounting for teachers in the Department" Information Systems ". The analysis of the existing rating system at the Department of Information Systems, developed UML diagrams and database AIS "Accounting for teachers in the Department" Information Systems "."

AIS "Accounting for teachers in the Department" Information Systems "" is designed to automate the calculation of the rating of teachers and head of the department of information systems, as well as to create a single database of the department.

Аңдатпа

Дипломдық жобада «Ақпараттық жүйелер» кафедрасынның оқытушыларының қызметін есептеу» автоматтандырылған ақпараттық жүйесін жасалынған. Ақпараттық жүйелер кафедрасында рейтингті есептеудің жүйелерін талдау жүзеге асырылған, UML диаграммалар мен «Ақпараттық жүйелер» кафедрасынның оқытушыларының қызметін есептеу» АЖЖ деректер қоры жасалынған.

«Ақпараттық жүйелер» кафедрасынның оқытушыларының қызметін есептеу» АЖЖ Ақпараттық жүйелер кафедрасынның оқытушылары мен меңгерушісінің рейтингісін есептеуді автоматтандыру, сонымен қатар кафедраның бірыңғай деректер қорын құру мақсатында жасалынған.

СОДЕРЖАНИЕ

Введение	9
1 Анализ предметной области	10
1.1 Структурные методы анализа и проектирования ИС	10
1.2 Анализ методики оценки качества работы преподавателей и заведующего кафедры	12
1.3 Обзор используемых программных и инструментальных средств	25
2 Разработка и проектирование АИС «Учет деятельности преподавателей кафедры «Информационные системы»	29
2.1 Содержательная постановка задачи	29
2.2 Анализ и проектирование АИС «Учет деятельности преподавателей кафедры «Информационные системы»	31
2.3 Проектирование базы данных	38
3 Программная реализация основных функциональных модулей системы	40
3.1 Разработка логики работы программ и пользовательских интерфейсов	40
3.2 Разработка алгоритмов обработки запросов пользователей	41
3.3 Описание назначения и функций программ, режимов работы	42
3.4 Описание категорий пользователей программы, разграничения прав доступа и разделение привилегий	43
3.5 Реализованные части программы	44
3.6 Проектирование пользовательского интерфейса	44
4 Техничко-экономическое обоснование	54
4.1 Расчет трудоемкости разработки Автоматизированной системы	55
4.2 Расчет затрат на разработку АИС	54
4.3 Определение возможной (договорной) цены ПП	61
5 Безопасность жизнедеятельности	63
5.1 Анализ опасных и вредных производственных факторов	63
5.2 Защитные мероприятия	67
Заключение	73
Список использованной литературы	74
ПРИЛОЖЕНИЕ А Техническое задание	76
ПРИЛОЖЕНИЕ Б Код программы	93

Введение

Рейтинг является одним из важнейших критериев оценки деятельности преподавателей высшего учебного заведения. Этот показатель оценки деятельности преподавателей влияет на заработную плату (оклад и надбавка), которая должна соответствовать результатам и качеству его работы. В данном дипломном проекте рассматривается расчет рейтинга как для преподавателей, так и для заведующего кафедры. Выбранная тема является актуальной по нескольким параметрам. Во-первых, на данный момент в Алматинском Университет Энергетики и Связи нет программы, которая могла бы рассчитывать рейтинг. Во-вторых, раз мы живем в век инноваций и компьютерных технологий, задача перевода ручного труда на вычислительные машины особенно важна.

Расчет рейтинга преподавателя производится для определения уровня его работы по многим параметрам. На основании рассчитанного рейтинга преподаватель занимает определенное место в списке преподавателей кафедры, а уже на основании этого ему может быть назначена надбавка к зарплате. Поэтому очень важно, чтобы рейтинг был рассчитан правильно, без ошибок.

Предметной областью данного дипломного проекта является кафедра информационных систем Алматинского университета энергетики и связи.

Для того, чтобы расчет производился правильно, существует ряд пунктов, которые необходимо выполнить. Итоговый рейтинг суммируется из коэффициентов. Информация для расчета этих коэффициентов поступает от самого преподавателя, заведующего кафедрой, других преподавателей кафедры, а так же секретаря кафедры и УМО.

Чтобы рейтинг был рассчитан без ошибок, необходима автоматизированная система расчета, которая поможет облегчить работу по подсчету всех коэффициентов. Автоматизация самого расчета приведет к тому, что сократится круг лиц, осуществляющих подсчет рейтинга. Преподаватель сможет отследить, по всем ли параметрам, а главное, правильно ли, подсчитали его рейтинг. Данная автоматизированная система так же будет способствовать экономии времени.

АИС «Учет деятельности преподавателей кафедры «Информационные системы» предназначена для комплексного подсчета рейтинга преподавателей кафедры Информационных систем.

1 Анализ предметной области

1.1 Структурные методы анализа и проектирования ИС

Анализ требований разрабатываемой системы является важнейшим среди всех этапов жизненного цикла. Он оказывает существенное влияние на все последующие этапы, являясь в то же время наименее изученным и понятным процессом. На этом этапе, во-первых, необходимо понять, что предполагается сделать, а во-вторых, задокументировать это, т.к. если требования не зафиксированы и не сделаны доступными для участников проекта, то они вроде бы и не существуют. При этом язык, на котором формулируются требования, должен быть достаточно прост и понятен заказчику.

Во многих аспектах системный анализ показывает наиболее тяжелой частью разработки. Нижеследующие проблемы, с которыми сталкивается системный аналитик, взаимосвязаны (и это является одной из главных причин их трудноразрешимости):

- аналитику сложно получить исчерпывающую информацию для оценки требований к системе с точки зрения заказчика;
- заказчик, в свою очередь, не имеет достаточной информации о проблеме обработки данных для того, чтобы судить, что является выполнимым, а что нет;
- аналитик сталкивается с чрезмерным количеством подробных сведений, как о предметной области, так и о новой системе;
- спецификация системы из-за объема и технических терминов часто непонятна для заказчика;
- в случае понятности спецификации для заказчика, она будет являться недостаточной для проектировщиков и программистов, создающих систему.

Конечно, применение известных аналитических методов снимает некоторые из перечисленных проблем анализа, однако эти проблемы могут быть существенно облегчены за счет применения современных структурных методов, среди которых центральное место занимают методологии структурного анализа.

Структурным анализом принято называть метод изучения системы, которое возникает с ее общего обзора и затем детализируется, приобретая иерархическое строение со все большим числом уровней. Для таких методов свойственно разбиение на уровни абстракции с ограничением числа элементов на каждом из уровней (обычно от 3 до 6-7); ограниченный контекст, включающий лишь существенные на каждом уровне детали; дуальность данных и операций над ними; использование строгих формальных правил записи; последовательное приближение к конечному результату.

Все методологии структурного анализа базируются на ряде общих принципов, часть из которых регламентирует организацию работ на начальных этапах жизненного цикла, а часть используется при выработке

рекомендаций по организации работ. В качестве двух базовых принципов используются следующие: принцип "разделяй и властвуй" и принцип иерархического упорядочивания. Первый является принципом решения трудных проблем путем разбиения их на множество меньших независимых задач, легких для понимания и решения. Вторым принципом декларируется, что устройство этих частей также существенно для понимания. Понимаемость проблемы резко повышается при организации ее частей в древовидные иерархические структуры, т.е. система может быть понята и построена по уровням, каждый из которых добавляет новые детали.

Однако стоит отметить, что остальные принципы структурного анализа являются не менее важными. Отметим основные принципы:

а) принцип абстрагирования состоит в выделении существенных с некоторых позиций аспектов системы и в отвлечении от несущественных с целью представления проблемы в простом общем виде;

б) принцип формализации заключается в необходимости строгого методического подхода к решению проблемы;

в) принцип упрятывания заключается в сокрытии несущественной на конкретном этапе информации: каждая часть "знает" только необходимую ей информацию;

г) принцип концептуальной общности заключается в следовании единой философии на всех этапах жизненного цикла (структурный анализ - структурное проектирование - структурное программирование - структурное тестирование);

д) принцип полноты заключается в контроле на присутствие лишних элементов;

е) принцип непротиворечивости заключается в обоснованности и согласованности элементов;

ж) принцип логической независимости заключается в концентрации внимания на логическом проектировании для обеспечения независимости от физического проектирования;

з) принцип независимости данных заключается в том, что модели данных должны быть проанализированы и спроектированы независимо от процессов их логической обработки, а также от их физической структуры и распределения;

и) принцип структурирования данных заключается в том, что данные должны быть структурированы и иерархически организованы;

к) принцип доступа конечного пользователя заключается в том, что пользователь должен иметь средства доступа к базе данных, которые он может использовать непосредственно (без программирования);

Соблюдение указанных принципов необходимо при организации работ на начальных этапах жизненного цикла независимо от типа разрабатываемого ПО и используемых при этом методологий.

1.2 Анализ методики оценки качества работы преподавателей и заведующего кафедры

На данный момент расчет рейтинга преподавателей и заведующего кафедры производится вручную с использованием инструментария Microsoft Office Excel. В связи с этим возможны ошибки из-за человеческого фактора.

Расчет рейтинга осуществляется согласно документу «Методика оценки качества работы (определение рейтинга) профессорско-преподавательского состава некоммерческого АО «АУЭС».

Основными положениями при расчете рейтинга являются:

– в соответствии с современными отношениями в экономике заработная плата (оклад и надбавка) каждого преподавателя должна соответствовать результатам и качеству его работы;

– администрация университета устанавливает надбавку к должностному окладу преподавателя за счет собственных средств университета по результатам его работы;

– надбавка к должностному окладу профессорско-преподавательского состава зависит от комплексного коэффициента R (рейтинга), определяемого для каждого преподавателя. Этот коэффициент включает несколько составляющих (K_1, K_2, \dots, K_7).

После определения коэффициентов K для всех преподавателей кафедр, составляется список преподавателей каждой должностной категории для всего университета, причем, место конкретного преподавателя в этом списке зависит от значения коэффициента R (рейтинга).

Величина надбавки к должностному окладу преподавателя зависит от места его в списке данной должностной категории, составленном по убыванию значения коэффициента R .

Определение комплексного коэффициента R (рейтинга)

Комплексный коэффициент R определяется из следующего выражения:

$$R = K_1 + K_2 + K_3 + K_4 + K_5 + K_6 + K_7, \quad (1.1)$$

Составляющие $K_1, K_2 \dots$ и т.д. определяется следующим образом.

Коэффициент K_1 . K_1 учитывает методическую работу преподавателя и определяется на основе результатов работы его за последние 2 календарных года. Вычисления проводятся самим преподавателем, утверждаются заведующим кафедрой и согласовываются с УМО.

$$K_1 = 1 + \frac{\text{Сумма баллов (п.1. таблицы 1)}}{20}, \quad (1.2)$$

Коэффициент K_2 . K_2 учитывает оценку труда данного преподавателя остальными преподавателями кафедры. Заведующий кафедрой (1 раз в конце семестра) подготавливает бланки для анонимного анкетирования (на бланке

указывается название кафедры и все фамилии преподавателей). Заполненные бланки опускаются преподавателями в специальную урну, установленную в УМО. Вычисления K2 проводит УМО.

$$K2 = \frac{\text{Сумма баллов (п.2 Таблицы 1)}}{20 \times N}, \quad (1.3)$$

где N – число анкет.

Коэффициент K3. K3 учитывает оценку труда преподавателем в области науки (см. таблица 1.3). Вычисления проводятся самим преподавателем и утверждаются начальником НИС. Коэффициент обновляется два раза в год. Выполненные работы учитываются в течении года, т.е. при проведении двух рейтинговых оценок.

Коэффициент K4. K4 учитывает оценку труда преподавателя с точки зрения декана (1 раз в конце каждого семестра) и устанавливается в диапазоне от 0 до 2. Значение K4 зависит от участия преподавателя в общественной жизни факультета, работы со студентами в общежитии, организации культурно-воспитательных мероприятий.

Коэффициент K5. K5 учитывает оценку труда преподавателя студентами. K5 равен среднему балу преподавателя (с округлением до первого знака после запятой) на основе анонимного анкетирования студентов, проведенного деканатами и УМО в предыдущем семестре. При оценке менее 3,0 коэффициент K5 не учитывается.

Коэффициент K6. K6 учитывает оценку труда преподавателей кафедры физического воспитания при подготовке призеров городских, республиканских соревнований, а также результаты сдачи студентами специальных тестов. K6 определяется на основе результатов работы его за последний календарный год. Баллы определяются заведующим кафедрой физического воспитания на основе соответствующих грамот, протоколов соревнований, результатов сдачи тестов.

$$K6 = \frac{\text{Баллы}}{20}, \quad (1.4)$$

Коэффициент K7. K7 учитывает работу преподавателя университета в качестве эдвайзера студенческой группы и заместителя декана по воспитательной работе в общежитии и устанавливается деканом факультета. Диапазон изменения коэффициента K7 устанавливается от 0 до 2 с шагом 0,5. K7 учитывает работу преподавателя в предыдущем семестре.

Таблица 1.1 – Оценка видов работы преподавателя для определения коэффициента надбавки к его должностному окладу

Методические материалы, постановка новых лабораторных работ (коэффициент K1)	Баллы
------------------------------------------------------------------------------	-------

Продолжение таблицы 1.1

Один учебник	80
Одно учебное пособие	60
Один конспект лекций	40
Участие в разработке ГОСО, типовых программ дисциплин, рабочих учебных планов	40
Одно методическое указание	25
Постановка новой лабораторной работы* – Физическая установка; – Виртуальная работа.	80 40
Методические материалы не представленные в течение года, соответствующего плану издания** – Одно методическое указание; Одно учебное пособие.	минус 12 минус 30

*если стенд (компьютерная программа) позволяет выполнить 2 или более работ, вводится поправочный коэффициент, увеличивающий пропорционально баллы п.1.6 (по согласованию с УМО).

**уменьшение баллов за методическую работу осуществляется каждый раз при определении рейтинга преподавателя до момента сдачи методических материалов в печать.

Таблица 1.2 – Взаимные оценки преподавателей внутри одной кафедры (коэффициент К2)

2.1 Профессиональные качества	до 10 баллов;
2.2 Педагогические качества	до 10 баллов;
2.3 Оценка заведующего кафедрой*	до 20 баллов;

*оценка проставляется с учетом опозданий и срывов занятий, допущенных преподавателем, зафиксированных УМО.

За проведенную работу в области науки преподавателю начисляются баллы, по которым определяется коэффициент К3. Коэффициент обновляется два раза в год. Выполненные работы учитываются в течение года.

Таблица 1.3 – Оценка труда преподавателя в области науки (коэффициент К3)

Баллы присуждаются за следующие виды работ	Подтверждающий документ	Баллы
--------------------------------------------	-------------------------	-------

Продолжение таблицы 1.3

<p>Руководитель темы (при наличии в составе авторов вышеназванных работ) в течение года после ее выполнения:</p> <ul style="list-style-type: none"> – бюджетные программы министерств и других центральных органов управления и регулирования Республики Казахстан; – хоздоговорные НИР и ОКР; – не финансируемые обязательные НИР ППС. 	<p>Защищенный отчет рук. темы</p>	<p>60 35 25</p>
<p>Исполнитель темы в течение года после ее выполнения</p> <ul style="list-style-type: none"> – бюджетные программы министерств и других центральных органов управления и регулирования Республики Казахстан – хоздоговорные НИР и ОКР – не финансируемые обязательные НИР ППС <p>Выполненными работами считаются: по финансируемым темам – подписанный Акт сдачи-приемки; по не финансируемым обязательным НИР ППС – работы, защищенные руководителями тем на НТС университета и представленные в виде отчета, в котором необходимо отразить:</p> <ul style="list-style-type: none"> – количество поданных заявок для участия в научных конкурсах и в тендерах на финансирование; – количество поданных заявок на авторское свидетельство, на изобретение (п/патент, патент), на открытие; – методики в области содержания и новых технологий в сфере образования; – количество поданных статей, докладов на конференции; – созданной лабораторной работы, программно-методического комплекса для совершенствования обучения, самообучения и дистанционного обучения с результатами внедрения научных разработок. 		<p>30 20 15</p>
<p>За защиту докторской диссертации;</p>	<p>докторская диссертация</p>	<p>75</p>
<p>За защиту докторской PhD диссертации;</p>	<p>PhD диссертация</p>	<p>60</p>
<p>За защиту кандидатской диссертации;</p>	<p>кандидатская диссертация</p>	<p>50</p>
<p>За научное руководство кандидатской диссертации;</p>	<p>Автореферат</p>	<p>50</p>

Продолжение таблицы 1.3

	диссертации	
За научное консультирование по докторской диссертации	Автореферат диссертации	50
За научное консультирование по докторской PhD диссертации	Автореферат диссертации	50
За опубликованную научную монографию после публикации (при коллективном авторстве сумма баллов распределяется в соответствии с долевым участием):	Монография	
– опубликованную в республиканских изданиях;		60
– опубликованную в зарубежных изданиях;		80
– опубликованную в дальнем зарубежье.		125
За получение охранного документа на изобретение (инновационный патент, патент, свидетельство на авторское право, диплом на открытие):	Копия диплома на открытие, патент, свидетельств о на авторское право, зарегистрированное в Казахстане, в России, в международных органах по фиксации интеллектуальной собственности	
– диплом на открытие;		150
– инновационный патент, патент, свидетельство на авторское право зарегистрированный в Казахстане;		35
– зарегистрированный в России;		60
– зарегистрированный в международных органах по фиксации интеллектуальной собственности.		70
За каждую статью, в течение года после публикации (при коллективном авторстве сумма баллов распределяется в соответствии с долевым участием):	Копии статей	
– опубликованную в республиканских изданиях, не входящих в Перечень ККОН (ВАК) РК;		10
– опубликованную в странах ближнего зарубежья;		25
– опубликованную в странах дальнего зарубежья.		30
– за каждую ссылку (цитируемость на статью в	Копии	3

Продолжение таблицы 1.3

других республиканских изданиях добавляется; за каждую ссылку на статью в международных изданиях добавляется.	титульного листа статьи, (монографии) и страницы со ссылкой	8
За каждый научный доклад сделанный на научной конференции, симпозиуме, конгрессе: – республиканский уровень; – международный уровень; – дальнее зарубежье.	Копии докладов	10 20 30
Работа в секциях научно-технических и научно-методических советов и комиссий при органах управления высшей школой, министерств и ведомств.	Приказ министерства, ведомства	40
Участие в работе конкурсных научно-технических комиссиях (действующих на постоянной основе): – факультетские; – университетские;	Выписки из протоколов заседания комиссии	5 15
Руководство научным семинаром (кружком) студентов (с периодичностью не менее 2-х раз в месяц) группой не менее 5 студентов	Выписки из протоколов семинара	25
Руководство НИРС группой до 4 студентов	Выписка из протокола заседания кафедры	20
Руководство призерами факультетских туров НИРС	Выписка из протокола факультетской комиссии	20
Руководство призерами университетских туров	Подтверждающий документ	35
Руководство призерами республиканских туров	Подтверждающий документ	45
Руководство призерами международных конкурсов	Подтверждающий документ	80
Руководство выпускными работами бакалавров – призерами республиканских и университетских конкурсов		45

* За невыполненную или незащищенную работу по обязательной НИР преподавателей университета начисляется отрицательный балл в размере, равном половине баллов по пунктам 1 и 2.

** При коллективном авторстве сумма баллов распределяется в соответствии с долевым участием.

Для расчета рейтинга заведующего кафедрой существует отдельный документ «Методика оценки качества работы (определение рейтинга) заведующих кафедрами некоммерческого АО «АУЭС»

Основные положения. Надбавка к должностному окладу зависит от комплексного коэффициента (рейтинга) R_3 определяемого для каждого заведующего кафедрой. Этот коэффициент включает несколько составляющих $K_{31}, K_{32}, \dots, K_{38}$.

По результатам расчета рейтинга составляются списки заведующих кафедрами отдельно для выпускающих и общеуниверситетских (невыпускающих) кафедр университета, причем, место конкретного заведующего кафедрой в этих списках зависит от значения коэффициента R_3

Величина надбавки к должностному окладу заведующего кафедрой зависит от места его в списке, составленном по убыванию значения коэффициента R_3 (рейтинга).

Определение комплексного коэффициента R_3 (рейтинга).

Комплексный коэффициент R_3 определяется из следующего выражения

$$R_3 = K_{31} + K_{32} + K_{33} + K_{34} + K_{35} + K_{36} + K_{37} + K_{38}, \quad (1.5)$$

где составляющие $K_{31}, K_{32}, \dots, K_{38}$ определяются следующим образом.

Коэффициент K_{31} . K_{31} учитывает выпуск и отставание от плана выпуска методических разработок кафедры и определяется на основе результатов работы кафедры за последний календарный год. Вычисления проводятся самим заведующим кафедрой и согласовываются с УМО (баллы определяются в соответствии с Таблицей 1.4. 1п.).

$$K_{31} = \frac{\text{Сумма баллов (п.1 Таблицы 1.4)}}{N}, \quad (1.6)$$

где N – число штатных единиц ППС кафедры, проработавших не менее одного года.

Для заведующего кафедрой физического воспитания при определении K_{31} сумма баллов определяется суммой коэффициента K_{36} преподавателей кафедры, учитывающем подготовку призеров соревнований, сдачу студентами специальных тестов и работу в специальных медицинских группах.

Коэффициент K_{32} . K_{32} учитывает оценку работы заведующего кафедрой преподавателям данной кафедры. Заведующий кафедрой (в конце каждого семестра) подготавливает бланки для анонимного анкетирования (на бланк указываются название кафедры и фамилия заведующего кафедрой).

Максимальное количество баллов – 40. Заполненные бланки опускаются преподавателями в специальную урну, установленную в УМО. Вычисления K_{32} проводит УМО

$$K_{32} = \frac{\text{Сумма баллов}}{20 \times N}, \quad (1.7)$$

где N – число штатных единиц ППС кафедры, проработавших не менее одного года.

Коэффициент K_{33} . K_{33} учитывает оценку работы преподавателей кафедры в области науки. Вычисления проводятся самими преподавателями и утверждаются начальником НИМ. Заведующий кафедрой определяет сумму коэффициентов в рамках вычисления коэффициента K_{33} всех преподавателей кафедры при определении своего рейтинга. Коэффициент обновляется в конце каждого семестра.

$$K_{33} = \frac{\text{Сумма коэффициентов } K_{33}(\text{каждого преподавателя})}{N} \quad (1.8)$$

где N – число штатных единиц ППС кафедры, проработавших не менее одного года.

Коэффициент K_{34} . K_{34} учитывает оценку исполнительской дисциплины и организаторской работы заведующего кафедрой и устанавливается отделом УМО, деканом и проректорами университета в диапазоне:

- УМО – от 0 до 2;
- декан – от 0 до 3;
- проректор по НР и МС – от 0 до 2;
- проректор по УМР – от 0 до 3.

Коэффициент обновляется в конце каждого семестра. Суммарное значение K_{34} зависит от участия кафедры в общественной жизни факультета, работы ППС со студентами в общежитии, организации культурно-воспитательных мероприятий, качества разработки учебной документации, выполнения научных разработок, выполнения поручений ректората. K_{34} определяется путем сложения вышеуказанных составляющих.

Коэффициент K_{35} . K_{35} учитывает оценку труда заведующего кафедрой со студентами в качестве преподавателя. K_{35} равен среднему баллу заведующего кафедрой (с округлением до первого знака после запятой) на основе анонимного анкетирования студентов, проведенного деканатами и УМО. Коэффициент обновляется в конце каждого семестра. При оценке менее 3,5 коэффициент K_{35} не учитывается.

Коэффициент K_{36} . K_{36} учитывает трудовую дисциплину сотрудников кафедры. K_{36} определяется на основе результатов работы кафедры за последний семестр. Баллы определяются заведующим кафедрой с учетом

срывов занятий, опозданий преподавателей на занятия или преждевременного окончания занятий (п.2, Таблица 1.6)

$$K_{36} = \frac{\text{Сумма баллов (п.2 таблицы 1.6)}}{N_{\text{общ}}}, \quad (1.9)$$

где $N_{\text{общ}}$ - общая численность ППС кафедры с учетом совместителей и почасовиков.

Коэффициент K_{37} . K_{37} учитывает работу преподавателей кафедры по внедрению новой техники и модернизации лабораторного оборудования в предыдущем семестре и оценивается УМО (см. таблица 1.7.п.3.). Этот коэффициент по желанию заведующего кафедрой может использоваться в течение года после внедрения (с разбивкой на части или один раз полностью).

$$K_{37} = \frac{\text{Сумма баллов (п.3 Таблицы 1.7)}}{2 \times N}, \quad (1.10)$$

где N – число штатных единиц ППС кафедры, проработавших не менее одного года.

Коэффициент K_{38} . K_{38} учитывает методическую и работу лично заведующего кафедрой. Коэффициент обновляется два раза в год. Выполненные работы учитываются в течение года, т.е. при проведении двух рейтинговых оценок.

$$K_{38} = 1 + \frac{\text{Сумма баллов за МР} + \text{Сумма баллов за НИР}}{40} \quad (1.11)$$

Сумма баллов за МР (методическую работу) определяется с применением Таблицы 1.1 Методики оценки качества работы (определения рейтинга ППС)

Сумма баллов за НИР (научную работу) определяется с применением Таблицы 1.4 Методики оценки качества работы (определения рейтинга) ППС.

* При коллективном авторстве в работах сумма баллов распределяется в соответствии с долевым участием.

**Уменьшение баллов за методическую работу осуществляется каждый раз при определении рейтинга заведующего кафедрой и определяется по срокам сдачи методических материалов в печать относительно плановых.

Таблица 1.4 - Оценка видов работы заведующего кафедрой для определения коэффициента надбавки к его должностному окладу

Методические материалы(коэффициент K_{31})	Баллы
Методические указания, выпущенные на кафедре (1 шт.)	5

Продолжение таблицы 1.4

Одно учебное пособие, выпущенное на кафедре	10
Один учебник, выпущенный на кафедре	20
Методические указания, не представленные в срок(1 шт.)	минус 7
Одно учебное пособие, не представленное в срок	минус 15

Таблица 1.5 – Трудовая дисциплина сотрудников кафедры (коэффициент К₃₆)

Число опозданий преподавателей кафедры на учебные занятия или преждевременное окончание занятий за семестр:	Баллы
1 случай	- 5
2 случая	- 15
3 случая	- 20
4 случая	- 30

Таблица 1.6 – Срыв учебных занятий

Срыв учебных занятий ППС кафедры	Баллы
1 срыв	- 20
2 срыва	- 30
3 срыва	- 40
4 срыва	- 50

Таблица 1.7 – Внедрение новой техники (коэффициент К₃₇)

Сдача в эксплуатацию одного стенда лабораторной работы (при наличии акта о сдаче в эксплуатацию)	Баллы
физическая модель	80
виртуальная модель	40

Для определения надбавок к окладам преподавателей и заведующего кафедры на основании рассчитанных коэффициентов ссылаются на документ «Методика определения надбавок к должностным окладам работников некоммерческого АО «АУЭС»

Основными положениями являются:

а) работникам устанавливаются персональные надбавки к должностным окладам за счет внебюджетных средств за высокие показатели в труде и добросовестное выполнение своих служебных обязанностей;

б) надбавки к должностным окладам для каждой категории работников рассчитывается в зависимости от финансового состояния университета, в т.ч. в первую очередь от поступления денег от студентов и магистрантов, обучающихся на платной основе;

в) для профессорско – преподавательского состава выпускающих кафедр при расчете надбавок устанавливается повышающий коэффициент в зависимости от доходности кафедр (за единицу отчета принимаются надбавки преподавателям общеуниверситетских кафедр);

г) для установления надбавок к должностным окладам работникам университета приказом ректора создается комиссия в составе проректоров, руководителей финансово – административного (ФАО) и учебно-методического (УМО) отделов. Председателем комиссии назначается один из проректоров университета;

д) комиссия вправе установить индивидуальную надбавку отдельным заведующим кафедрами, руководителям подразделений, преподавателям и другим работникам в случае выполнения ими некоторых особо важных (сложных, ответственных и т.п.) видов работ, не оговоренных в настоящей методике, а также: при переходе работника с руководящей должности на рядовую в связи с окончанием срока полномочий в должности руководителя; для завершения диссертационной работы; в связи с заграничной стажировкой и другим аналогичным уважительным причинам;

е) решение комиссии по установлению надбавок работникам университета оформляется протоколом и подписывается всеми членами комиссии;

На основании решения комиссии приказом по университету работникам университета устанавливаются надбавки к должностным окладам.

Порядок установления надбавок профессорско - преподавательскому составу:

а) в соответствии с «Методикой оценки качества работы профессорскопреподавательского состава» два раза в течение учебного года (перед началом каждого семестра) рассчитывается рейтинг преподавателя R по каждой должностной группе (профессора; доценты; старшие преподаватели; преподаватели и ассистенты);

б) комиссия университета по установлению надбавок в соответствие с нормальным законом распределения рейтинга преподавателей исключает из рейтинга-листа преподавателей, имеющих рекордный (сверхвысокий) рейтинг.

Указанные преподаватели автоматически относятся к преподавателям, имеющим наивысшую величину надбавки (1 категория) к должностным окладам;

в) в каждой должностной группе определяется разница между максимальной (R_{max}) и минимальной (R_{min}) величиной рейтинга. За начальную точку отсчета принимается минимальный рейтинг преподавателя соответствующей группы;

г) указанная разница (превышение над минимальным рейтингом) разбивается на 4 категории, определяющие величину надбавки к должностным окладам ППС). Граничные значения величины рейтингов,

соответствующих каждой категории надбавки, подсчитываются по формулам:

- 1) I категория $RI \geq (R_{min}) + 0,7(R_{max} - R_{min})$;
- 2) II категория $R_{II} \geq (R_{min}) + 0,2(R_{max} - R_{min}) < RI$;
- 3) III категория $R_{III} \geq (R_{min}) + 0,05 (R_{max} - R_{min}) < R_{II}$;
- 4) IV категория $R_{IV} < R_{III}$.

Преподаватели, имеющие первые три категории рейтинга отвечают требованиям университета. Преподаватели, имеющие IV категорию рейтинга, не в полной мере соответствуют предъявляемым требованиям;

д) в рейтинг-листе по каждой должностной группе наносятся грани внутриуниверситетских категорий по надбавке и определяется пофамильно список преподавателей, претендующих на ту или иную величину доплаты;

е) в соответствии с таблицей абсолютных величин надбавок устанавливается надбавка к должностному окладу каждого преподавателя:

1) отдельным категориям преподавателей дополнительно устанавливается доплата за работу эдвайзером группы, заместителем декана по воспитательной работе в общежитии, выполнение обязанностей ответственных в подразделениях, делопроизводство на государственной языке;

2) преподавателям, имеющим средний балл менее 3,0 и замечания по результатам студенческого анкетирования, категория надбавки снижается, как минимум, на одну позицию.

Размер надбавок работникам по совмещаемой педагогической должности устанавливается:

– для штатных преподавателей, ведущих педагогическую работу по внутривузовскому совместительству (в зависимости от финансового состояния университета) к окладу по совмещаемой должности – не менее 50%, но не более размера надбавки, установленной по основной должности;

– для остальных работников университета – по II категории надбавки соответствующей должностной группы для лиц с учеными степенями и званиями и по III категории для лиц, не имеющих ученых степеней и званий;

– для вновь принимаемых совместителей со стороны – по II категории надбавки соответствующей должностной группы для лиц с учеными степенями и званиями и по III категории для лиц, не имеющих ученых степеней и званий;

– для совместителей со стороны со стажем работы в университете не менее одного года - как правило, по II категории надбавки соответствующей должностной группы.

Для вновь принимаемых на работу преподавателей устанавливается надбавка:

– с ученой степенью – по II категории надбавки соответствующей должностной группы;

– без ученой степени – по III категории надбавки соответствующей должностной группы.

Данная надбавка сохраняется для вновь принятых преподавателей в течение двух семестров. В этом периоде величина надбавки может измениться, если:

– преподаватель по результатам очередного расчета рейтинга получил более высокую категорию надбавки;

– преподаватель получил средний балл ниже 3,0 и /или имеет замечания по результатам анкетирования студентов (категория надбавки снижается, как минимум, на одну позицию).

Порядок установления надбавок заведующим кафедрами:

а) в соответствии с «Методикой оценки качества работы (определения рейтинга) заведующих кафедрами» два раза в течение учебного года (перед началом каждого семестра) рассчитывается рейтинг заведующих кафедрами;

б) по результатам расчета рейтинга составляются списки заведующих кафедрами отдельно для выпускающих и невыпускающих кафедр университета.

в) по каждой группе кафедр устанавливаются 3 категории надбавки к должностным окладам:

1) I категория – 20% заведующих кафедрами, имеющих наивысший рейтинг;

2) III категория – 10% заведующих кафедрами, имеющих наименьший рейтинг. Остальным заведующим кафедрами устанавливается II категория надбавки к должностному окладу.

г) заведующими кафедрами может быть снижена категория надбавки как минимум на одну позицию в случае, если по результатам студенческого анкетирования:

1) имеются замечания к ППС;

2) средний балл заведующего кафедрой менее 3,5;

3) отсутствует методическое обеспечение по дисциплинам кафедры.

Порядок установления надбавок другим категориям работников:

а) остальным категориям работников (административный, учебно-вспомогательный и обслуживающий персонал, в том числе совместителям студентам/магистрантам и совместителям со стороны) два раза в течение учебного года определяется средняя величина надбавки в процентном отношении от величины должностного оклада по каждой категории;

б) надбавка к должностному окладу устанавливается каждому сотруднику по предложению руководителя подразделения с учетом трудовой дисциплины и качества исполнения служебных обязанностей. Отдельным работникам к надбавке устанавливается доплата за выполнение обязанностей ответственных по подразделению–делопроизводство на государственном языке;

в) для вновь принимаемых работников, указанных в п. 4.1, на период момента установления новых персональных надбавок устанавливается

надбавка равная половине средней величины действующей надбавки по каждой категории работников.

В порядке исключения проректора по принадлежности можно устанавливать принимаемым работникам индивидуальные надбавки к должностным окладам.

Порядок установления надбавок работникам, обслуживающим учебный процесс в магистратуре.

Для ППС и УВП, обслуживающих учебный процесс магистрантов устанавливается дополнительная надбавка к должностному окладу, равная

$$H_{\text{маг}} = O_{\text{д}} \times K_{\text{маг}} \times s \times \left(\frac{N_{\text{маг}}}{N_{\text{общ}}} \right), \quad (1.12)$$

где $H_{\text{маг}}$ – надбавка в тенге к должностным окладам ППС и УВП, устанавливая на учебный год с корректировкой этой величины в январе месяце на весь семестр;

$O_{\text{д}}$ – должностной оклад;

$K_{\text{маг}}$ – коэффициент, величина которого устанавливается комиссией по установлению надбавок работникам по категориям: доктора наук, кандидаты и/или доценты ВАК, ППС без степени, УВП;

s – количество занимаемых ставок работника (0,25 – 1,5);

1.3 Обзор используемых программных и инструментальных средств

Средства структурного анализа и их взаимоотношения. Для целей моделирования систем вообще, и структурного анализа в частности, используются три группы средств, иллюстрирующих:

- функции, которые система должна выполнять;
- отношения между данными;
- зависящее от времени поведение системы (аспекты реального времени).

Среди всего многообразия средств решения данных задач в методологиях структурного анализа наиболее часто и эффективно применяемыми являются следующие:

- DFD (Data Flow Diagrams) - диаграммы потоков данных совместно со словарями и спецификациями процессов или миниспецификациями;
- ERD (Entity-RElationship Diagrams) - диаграммы "сущность-связь";
- STD (StatE Transition Diagrams) - диаграммы переходов состояний.

Все они содержат графические и текстовые средства моделирования: первые - для удобства демонстрирования основных компонент модели, вторые - для обеспечения точного определения ее компонент и связей.

Логическая DFD показывает внешние по отношению к системе источники истоки (адресаты) данных, идентифицирует логические функции (процессы) и группы элементов данных, связывающие одну функцию с

другой (потoki), а также идентифицирует хранилища (накопители) данных, к которым осуществляется доступ. Структуры потоков данных и определения их компонент хранятся и анализируются в словаре данных. Каждая логическая функция (процесс) может быть детализирована с помощью DFD нижнего Уровня; когда дальнейшая детализация перестает быть полезной, переходят к выражению логики функции при помощи спецификации процесса (миниспецификации). Содержимое каждого хранилища также сохраняют в словаре данных, модель данных хранилища раскрывается с помощью ERD. В случае наличия реального времени DFD дополняется средствами описания зависящего от времени поведения системы, раскрываемыми с помощью STD. Эти связи показаны на рисунке 1.

Перечисленные средства дают полное описание системы независимо от того, является ли она существующей или разрабатываемой с нуля. Таким образом строится логическая функциональная спецификация - подробное описание того, что должна делать система, освобожденное насколько это возможно от рассмотрения путей реализации. Это дает проектировщику четкое представление о конечных результатах, которые следует достигать.

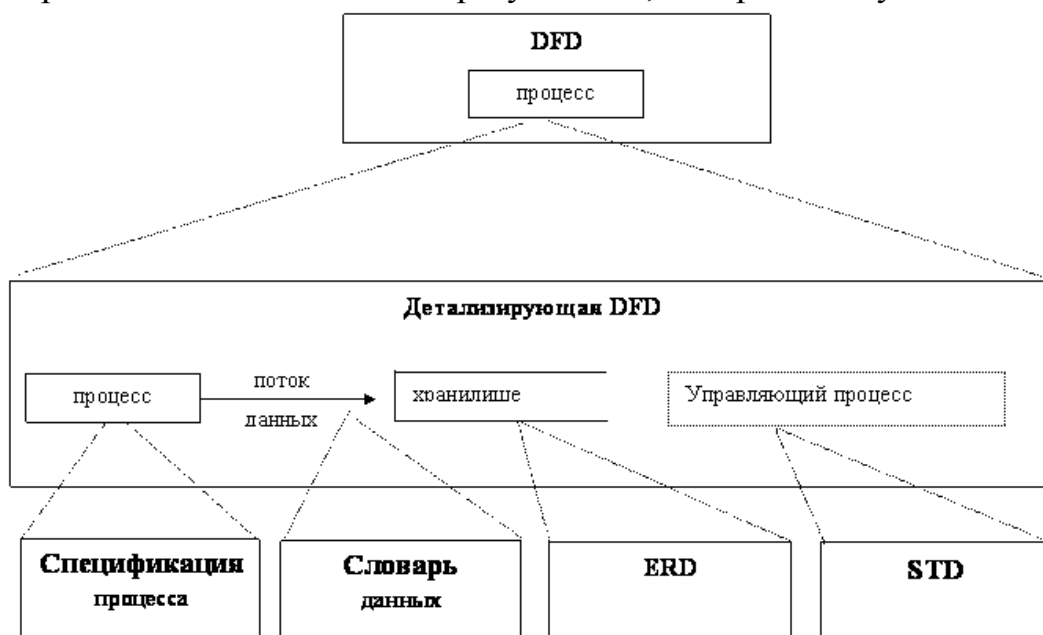


Рисунок 1.1 - Компоненты логической модели

Объектно-ориентированные методы анализа и проектирования ИС. Объектно-ориентированная технология широко применяется при разработке программного обеспечения больших информационных систем, но она до сих пор находится на стадии становления. Особенно это касается первой стадии процесса разработки – анализа предметной области. Большие, сложные системы в принципе не поддаются формальному описанию. При анализе предметная область разделяется на объекты, обладающие определёнными свойствами и вступающие во взаимодействие между собой. Поиск «правильных» категорий объектов требует больших усилий, при этом само

понятие «правильности» не удаётся чётко сформулировать. Очень непросто выработать устойчивую систему абстракций, определяющую соответствующие объекты, описать их взаимодействие. При этом надо предвидеть сложности реализации объектов, обеспечивающей их повторное использование при проектировании других систем.

Именно повторное использование и простота модификации гарантирует эффективность объектно-ориентированного программирования.

Многие исследователи разрабатывали каждый свою методику объектно-ориентированного анализа предметной области.

Современные методики объектно-ориентированного анализа. Различные методики объектно-ориентированного анализа имеют общие черты, но отличаются акцентами на определённые аспекты процесса моделирования предметной области. Общая терминология обуславливает взаимопроникновение идей и понятий всех методов. В результате объединения этих методик были созданы стандарт описания моделей IDEF и язык моделирования UML.

Основная процедура объектно-ориентированного анализа – построение информационной модели, в которой фундаментальная структура предметной области абстрагируется от реального мира и чётко формализуется и документируется. Большинство исследователей выделяют три разновидности фундаментальной структуры предметной области.

Функциональная структура – упорядоченная совокупность действий, процессов и операций, производимых анализируемой системой. Компонентная структура состоит из элементов (объектов) и их взаимосвязей. Динамическая (поведенческая) структура описывает возможные состояния системы и события, вызывающие переход из одного состояния в другое. Они могут анализироваться все вместе и должны быть согласованы друг с другом.

Для отдельных областей применения ООА приобретают значение распределение объектов и процессов во времени и пространстве (например, при разработке систем реального времени) или структура ограничений и правил поведения и использования (например, для бизнес-процессов). Конечная цель объектно-ориентированного анализа – представить предметную область в терминах классов и объектов для дальнейшего проектирования логической структуры разрабатываемого программного обеспечения. Поэтому при анализе планируется наряду с вышеперечисленными концептуальная структура, объединяющая функции, компоненты, события, сценарии поведения с абстрактными понятиями, характерными для понимания сущности предметной области в иерархию классов. Средствами описания функциональной структуры являются контекстные диаграммы, диаграммы процессов и потоков (DFD), снабжённые спецификацией используемых имён. Отметим необходимость иерархической организации диаграмм по вложению для удобства представления большого количества объектов и связанную с этим проверку логической непротиворечивости составных частей. Эти трудоёмкие

процедуры лучше производить с использованием специальных компьютерных программ моделирования. Параллельно с функциональной структурой разрабатывается структура используемых данных. Известные диаграммы «сущностей-связей» (ERD) помогают отразить связи, невидимые на DFD, и дополняют их. В дальнейшем они используются для реализации в СУБД. Для отображения динамического поведения системы используются диаграммы переходов состояний (STD), диаграммы деятельности языка UML, которые позволяют выделить основные механизмы реагирования на события. Дальнейшее структурирование множества операций снова приводит к DFD, но уже более детализированным. Таким образом процесс анализа становится итеративным. После проверки полноты и непротиворечивости модели создаются предварительные диаграммы классов и объектов, служащие отправной точкой для следующего этапа разработки объектно-ориентированного проектирования[1].

CASE – средства визуального моделирования. За последние два десятилетия сформировалось новое направление в программной инженерии - CASE (Computer-Aided Software/System Engineering). Не существует общепринятого определения CASE. Содержание этого понятия обычно определяется перечнем задач, решаемых с помощью CASE-технологий. Очень грубо, CASE-технология представляет собой совокупность методологий анализа, проектирования, разработки и сопровождения сложных систем программного обеспечения, поддержанную комплексом взаимосвязываемых средств автоматизации. CASE - это инструментарий для системных аналитиков, разработчиков и программистов, который позволяет описывать бизнес-процессы на компьютере, используя полученные схемы при разработке или настройке системы.

В большинстве современных CASE-систем применяются методологии структурного анализа и проектирования, основанные на наглядных диаграммных техниках, при этом для описания модели проектируемой системы используются графы, диаграммы, таблицы и схемы. Такие методологии обеспечивают строгое и наглядное описание проектируемой системы, которое начинается с ее общего обзора и затем детализируется, приобретая иерархическую структуру со все большим числом уровней.

Помимо автоматизации структурных методологий и. Как следствие, возможности применения современных методов системной и программной инженерии, CASE обладают следующими достоинствами:

- улучшают качество создаваемого ПО за счет средств автоматического контроля (прежде всего, контроля проекта);
- позволяют за короткое время создать прототип будущей системы, что позволяет на ранних этапах оценить ожидаемый результат;
- ускоряют процесс проектирования и разработки;
- поддерживают развитие и сопровождение разработки;
- поддерживают технологии повторного использования компонент разработки.

Большинство CASE-средств основано на парадигме методология/метод/нотация/средство. Методология определяет руководящие указания для оценки и выбора проекта разрабатываемого ПО, шаги работы и их последовательность, а также правила распределения и назначения методов. Метод - это систематическая процедура или техника генерации описаний компонент ПО (например, проектирование потоков и структур данных). Нотации предназначены для описания структуры системы, элементов данных, этапов обработки и включает графы, диаграммы, таблицы, блок-схемы, формальные и естественные языки. Средства - инструментарий для поддержки и усиления методов. Эти инструменты поддерживают работу пользователей при создании и редактировании графического проекта в интерактивном режиме, они способствуют организации проекта в виде иерархии уровней абстракции, выполняют проверки соответствия компонент[2].

2 Разработка и проектирование АИС «Учет деятельности преподавателей кафедры «Информационные системы»

2.1 Содержательная постановка задачи

Назначение системы – расчет рейтинга преподавателей кафедры Информационных систем Алматинского университета энергетики и связи.

Цели создания системы – автоматизация, упрощение и сокращение времени расчета рейтинга преподавателей кафедры Информационных систем Алматинского университета энергетики и связи.

2.1.1 Описание предметной области

Предметной областью данного курсового проекта является кафедра информационных систем Алматинского университета энергетики и связи.

Для того, чтобы расчет производился правильно, существует ряд пунктов, которые необходимо выполнить. Итоговый рейтинг суммируется из коэффициентов. Информация для расчета этих коэффициентов поступает от самого преподавателя, заведующего кафедрой, других преподавателей кафедры, а так же секретаря кафедры и УМО.

Чтобы рейтинг был рассчитан без ошибок, необходима автоматизированная система расчета, которая поможет облегчить работу по подсчету всех коэффициентов. Автоматизация самого расчета приведет к тому, что сократится круг лиц, осуществляющих подсчет рейтинга. Преподаватель сможет отследить, по всем ли параметрам, а главное, правильно ли, подсчитали его рейтинг. Данная автоматизированная система так же будет способствовать экономии времени.

АИС «Учет деятельности преподавателей кафедры «Информационные системы» предназначена для комплексного подсчета рейтинга преподавателей кафедры Информационных систем.

2.1.2 Анализ требований

Бизнес-процесс — это совокупность взаимосвязанных мероприятий или задач, направленных на создание определенного продукта или услуги для потребителей. Для наглядности бизнес-процессы визуализируют при помощи блок-схемы бизнес-процессов.

Бизнес-процесс: расчет рейтинга:

- ввод данных о преподавателе;
- расчет коэффициентов;
- определение комплексного коэффициента.

Весь этот процесс осуществляется на основании положения университета.

2.1.3 Разработка технического задания

Техническое задание является исходным материалом для создания информационной системы или другого продукта. Поэтому техническое задание (сокращенно ТЗ) в первую очередь должно содержать основные технические требования к продукту и отвечать на вопрос, что данная система должна делать, как работать и при каких условиях.

Как правило, этапу составления технического задания предшествует проведение обследования предметной области, которое завершается созданием аналитического отчета. Именно аналитический отчет (или аналитическая записка) ложится в основу документа Техническое задание.

Если в отчете требования заказчика могут быть изложены в общем виде и проиллюстрированы UML-диаграммами, в техническом задании следует подробно описать все функциональные и пользовательские требования к системе. Чем подробнее будет составлено техническое задание, тем меньше спорных ситуаций возникнет между заказчиком и разработчиком во время приемочных испытаний.

Таким образом, техническое задание является документом, который позволяет как разработчику, так и заказчику представить конечный продукт и впоследствии выполнить проверку на соответствие предъявленным требованиям.

Обобщенная структура разделов технического задания:

- что надо сделать;
- зачем;
- где это будет работать;
- каким требованиям должно удовлетворять;
- какие работы потребуется выполнить, чтобы сделать это;
- каков порядок приемки-сдачи работ Заказчику;
- как должно быть задокументировано проведение работ;
- на основании каких нормативно-технических документов должны проводиться работы.

Техническое задание для АИС «Учет деятельности преподавателей кафедры «Информационные системы» представлено в ПРИЛОЖЕНИЕ А Техническое задание.

2.2 Анализ и проектирование АИС «Учет деятельности преподавателей кафедры «Информационные системы»

2.2.1 Вид с точки зрения поведения

На диаграмме прецедентов (вариантов использования) показано взаимодействие между вариантами использования и действующими лицами.

Она отражает требования к системе с точки зрения пользователя. Таким образом, варианты использования – это функции, выполняемые системой, а действующие лица – это заинтересованные по отношению к создаваемой системе.

Основная задача диаграммы вариантов использования - представлять собой единое средство, дающее возможность заказчику, конечному пользователю и разработчику совместно обсуждать функциональность и поведение системы. [4]

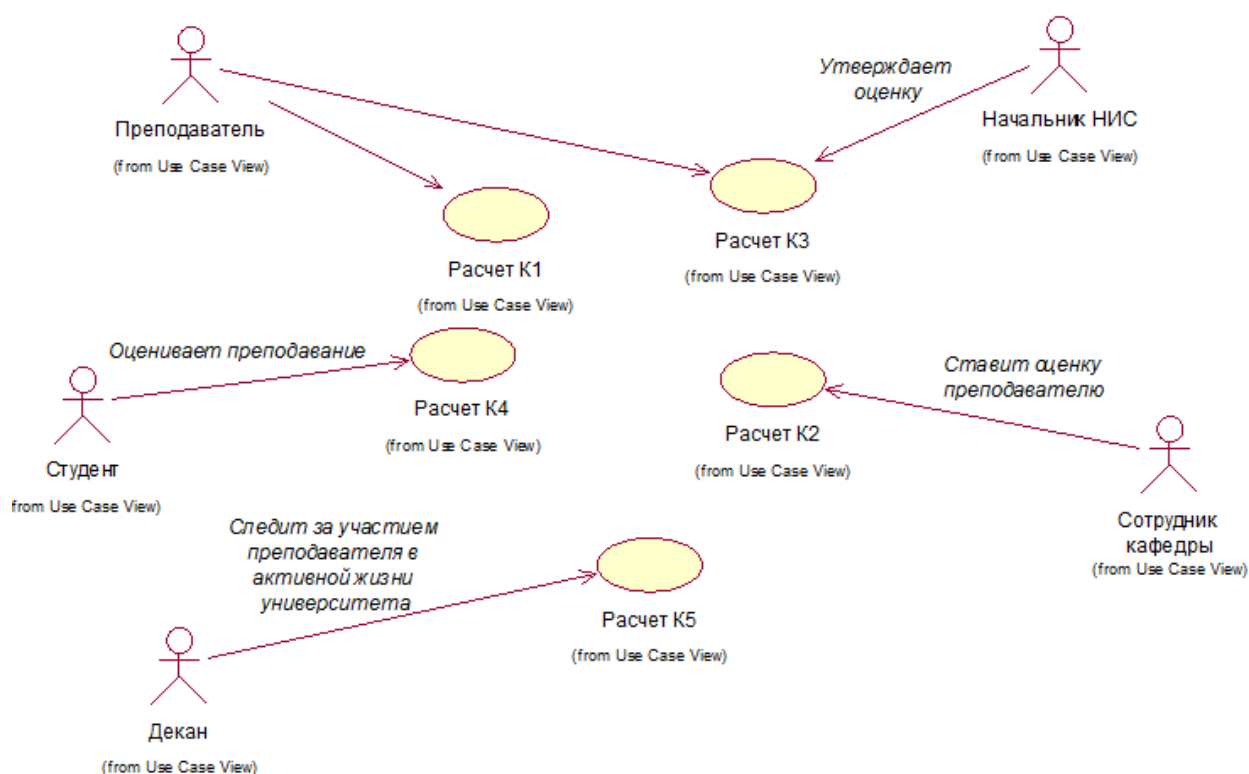


Рисунок 2.1 – Диаграмма прецедентов (as-is)

Актер (actor) - любая внешняя по отношению к проектируемой системе сущность, которая взаимодействует с системой и использует ее функциональные возможности для достижения определенных целей или решения частных задач.

В нашей работе пять актеров или пять действующих лиц: преподаватель, начальник НИС, сотрудник кафедры, декан и студент. Также существует 5 действий, предполагающих расчет рейтинга: расчет K1, расчет K2, расчет K3, Расчет K4 и расчет K5. На диаграмме наглядно видно, что у каждого действующего строго определенное действие.

Создаваемая АИС «Учет деятельности преподавателей кафедры «Информационные системы» предполагает уменьшение действующих лиц. Для расчетов коэффициентов K2, K4 и K5 нужен будет только секретарь кафедры, а начальник НИС, сотрудник кафедры, декан и студент становятся внешними источниками, от которых секретарь берет данные для заполнения таблицы.

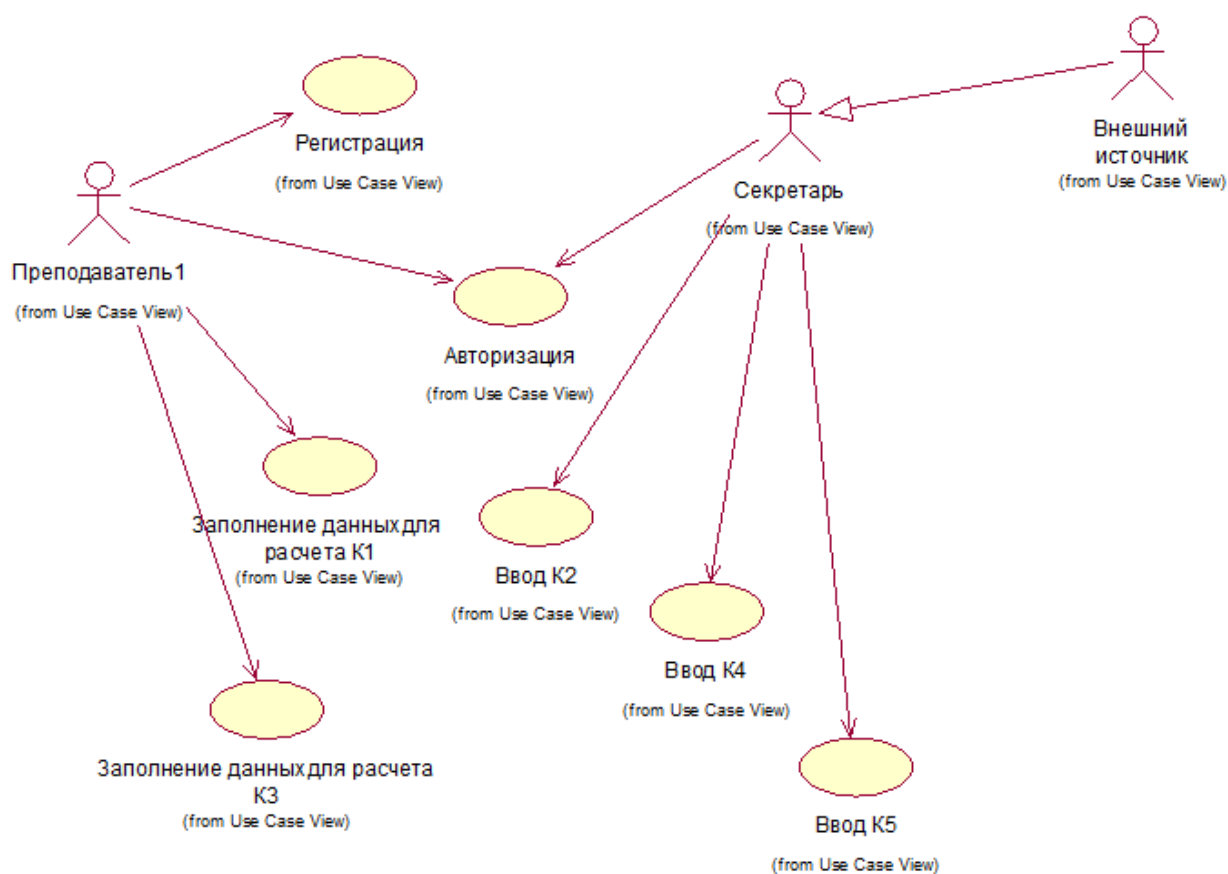


Рисунок 2.2 - Диаграмма прецедентов (to BE)

На этой диаграмме уже три актера: преподаватель, секретарь и внешний источник. Появились дополнительные действия: регистрация и авторизация.

Сценарий диаграммы прецедентов АИС «Учет деятельности преподавателей кафедры «Информационные системы»:

Вариант использования: Расчет рейтинга преподавателя.

Актеры: преподаватель, секретарь и внешний источник.

Цель: рассчитать рейтинг преподавателей кафедры.

Краткое описание: Преподаватель регистрируются, затем авторизуется. Преподаватель заполняет данные для расчета К1 и для расчета К3. Секретарь получает данные из внешнего источника. Секретарь регистрируется, затем авторизуется. Секретарь вводит К2, К4 и К5.

Тип: Базовый

2.2.2 Вид с точки зрения процесса

Несмотря на то, что деятельность кафедры предусматривает множество разнообразных действий исполнителей, для нашей задачи существенными являются только процессы обмена информацией между этими исполнителями, и именно они отображаются в создаваемых моделях. Поэтому на диаграмме отражен процесс формирования отчетности.

Общее поле диаграммы деятельности делится на несколько «плавательных дорожек», каждая из которых содержит описание действий одного из исполнителей.

Основными элементами диаграмм видов деятельности являются обозначения состояния («начало», «конец»), действия (овал) и момента синхронизации действий (линейка синхронизации, на которой сходятся или разветвляются несколько стрелок) [3].

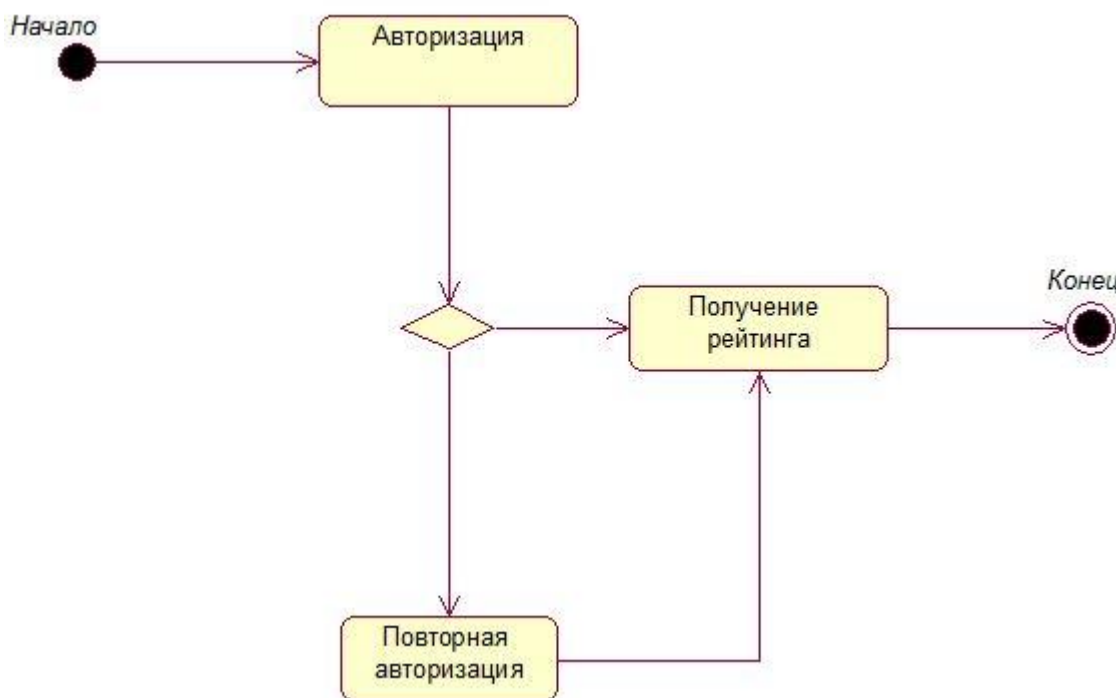


Рисунок 2.3 - Диаграмма видов деятельности

2.2.3 Вид с точки зрения проектирования

Диаграмма последовательности применяется для визуализации процесса взаимодействия объектов, а не как средство моделирования алгоритма управления. Это диаграмма, на которой показаны взаимодействия объектов, упорядоченные по времени их проявления.

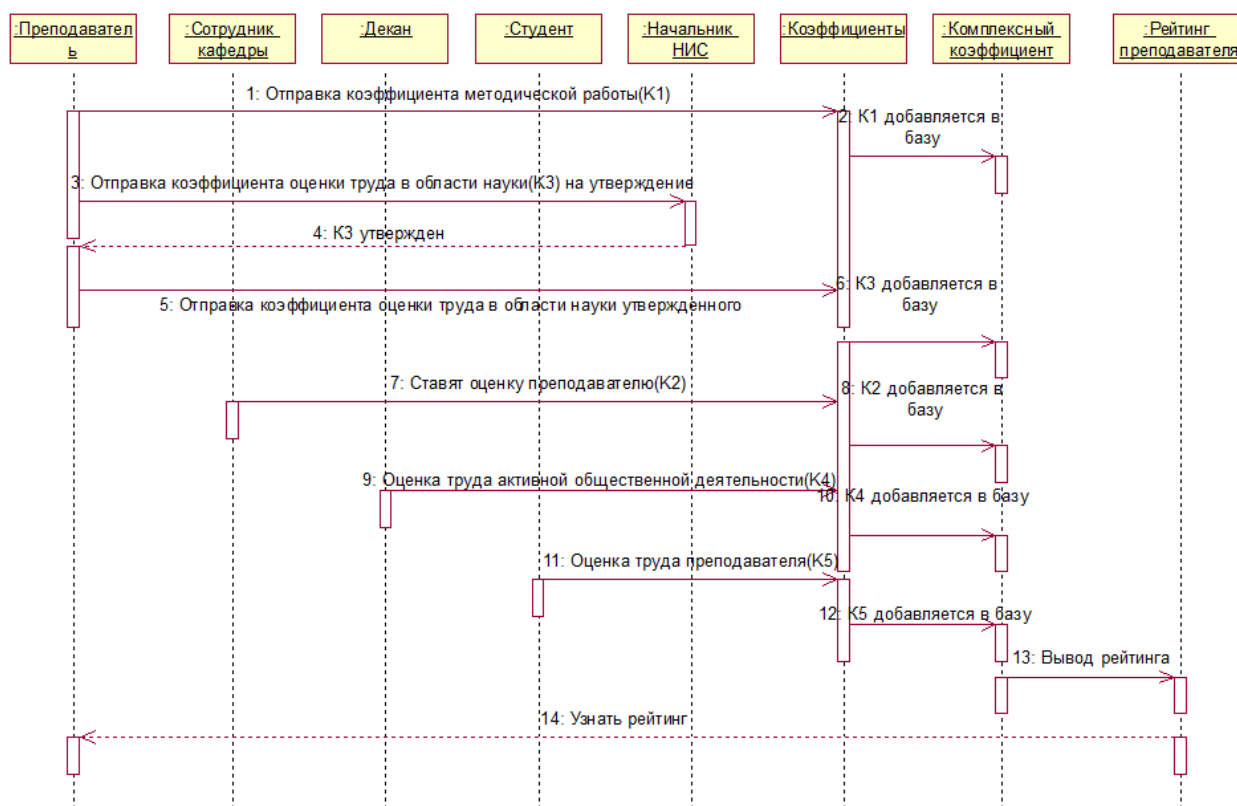


Рисунок 2.4 - Диаграмма последовательности

Разберем каждый элемент диаграммы, по отдельности:

Объект, Участник (Object, Participant)

Обозначается прямоугольником, в котором указывается информация об участнике действий. Это, как правило, название объекта и его класс, разделенные двоеточием. Располагаются объекты (как правило) вдоль верхнего края диаграммы. От прямоугольника вниз спускается Линия Жизни.

Линия жизни (Life Line)

Линия, идущая вниз от участника, обозначающая отведенное объекту время жизни. Обозначается пунктирной линией.

Активация, фрагмент выполнения (Activation Bar, Execution Occurrences)

Обозначается узким прямоугольником (серого или белого цвета), расположенным на линии жизни. Указывает начало и завершение действия, в котором участвует объект. Поскольку линия жизни - это метафора времени, то прямоугольник на линии жизни указывает на активизацию объекта во времени [4].

2.2.4 Вид с точки зрения реализации

Диаграммы состояний определяют все возможные состояния, в которых может находиться конкретный объект, а также процесс смены состояний объекта в результате наступления некоторых событий.

На диаграмме имеются два специальных состояния - начальное (start) и конечное (stop). Начальное состояние выделено черной точкой, оно соответствует состоянию объекта, когда он только что был создан. Конечное состояние обозначается черной точкой в белом кружке, оно соответствует состоянию объекта непосредственно перед его уничтожением. На диаграмме состояний может быть одно и только одно начальное состояние.

В данном курсовом проекте диаграмма состояний не нужна, так как все возможные состояния определены и они последовательны, в которых может находиться конкретный объект, а также процесса смены состояний объекта в результате наступления некоторых событий не происходит.

Для представления физических сущностей в языке UML применяется специальный термин - компонент (componEnt). Компонент реализует некоторый набор интерфейсов и служит для общего обозначения элементов физического представления модели. Для графического представления компонента может использоваться специальный символ - прямоугольник со вставленными слева двумя более мелкими прямоугольниками. Внутри объемлющего прямоугольника записывается имя компонента и, возможно, некоторая дополнительная информация. Изображение этого символа может незначительно варьироваться в зависимости от характера ассоциируемой с компонентом информации [6].

В метамодели языка UML компонент является потомком классификатора. Он предоставляет организацию в рамках физического пакета ассоциированным с ним элементам модели. Как классификатор, компонент может иметь также свои собственные свойства, такие как атрибуты и операции [4].

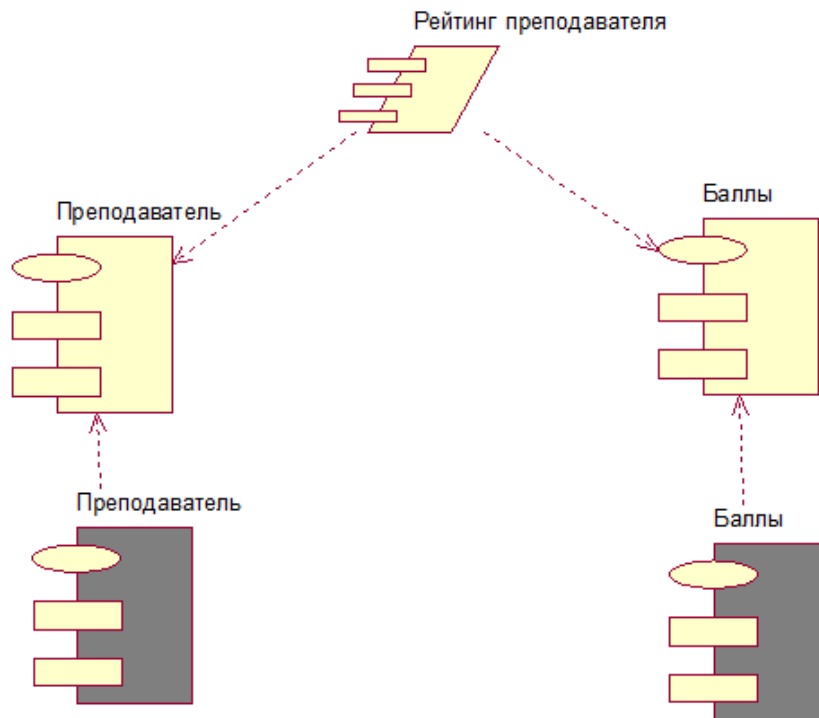


Рисунок 2.5 – Диаграмма компонентов

Главный компонент, который фактически управляет остальными - это Рейтинг преподавателя.

Каждый компонент состоит из двух частей:

а) спецификация – это заголовочный файл для сведений о прототипах функций для класса (не закрашенная часть);

б) тело пакета – часть, которая содержит код операции класса (закрашенная часть).

Диаграмма компонентов показывает, как выглядит модель на физическом уровне [6].

2.2.5 Вид с точки зрения логики

Диаграммы классов UML. Логическое моделирование. Класс – это основной строительный блок ПС. Это понятие присутствует и в ОО языках программирования, то есть между классами UML и программными классами есть соответствие, являющееся основой для автоматической генерации программных кодов или для выполнения реинжиниринга. Каждый класс имеет название, атрибуты и операции. Класс на диаграмме показывается в виде прямоугольника, разделенного на 3 области. В верхней содержится название класса, в средней – описание атрибутов (свойств), в нижней – названия операций – услуг, предоставляемых объектами этого класса.

Атрибуты класса определяют состав и структуру данных, хранимых в объектах этого класса. Каждый атрибут имеет имя и тип, определяющий, какие данные он представляет. При реализации объекта в программном коде

для атрибутов будет выделена память, необходимая для хранения всех атрибутов, и каждый атрибут будет иметь конкретное значение в любой момент времени работы программы. Объектов одного класса в программе может быть сколько угодно много, все они имеют одинаковый набор атрибутов, описанный в классе, но значения атрибутов у каждого объекта свои и могут изменяться в ходе выполнения программы.

Для каждого атрибута класса можно задать видимость (*visibility*). Эта характеристика показывает, доступен ли атрибут для других классов. В UML определены следующие уровни видимости атрибутов:

- открытый (*public*) – атрибут виден для любого другого класса (объекта);
- защищенный (*protected*) – атрибут виден для потомков данного класса;
- закрытый (*private*) – атрибут не виден внешними классами (объектами) и может использоваться только объектом, его содержащим.

Последнее значение позволяет реализовать свойство инкапсуляции данных. Например, объявив все атрибуты класса закрытыми, можно полностью скрыть от внешнего мира его данные, гарантируя отсутствие несанкционированного доступа к ним. Это позволяет сократить число ошибок в программе. При этом любые изменения в составе атрибутов класса никак не скажутся на остальной части ПС.

Класс содержит объявления операций, представляющих собой определения запросов, которые должны выполнять объекты данного класса. Каждая операция имеет сигнатуру, содержащую имя операции, тип возвращаемого значения и список параметров, который может быть пустым. Реализация операции в виде процедуры – это метод, принадлежащий классу. Для операций, как и для атрибутов класса, определено понятие «видимость». Закрытые операции являются внутренними для объектов класса и недоступны из других объектов. Остальные образуют интерфейсную часть класса и являются средством интеграции класса в ПС.

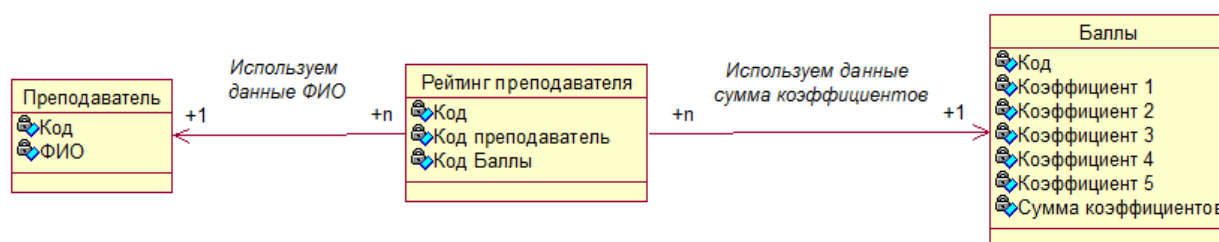


Рисунок 2.6 – Диаграмма классов

На диаграммах классов обычно показываются ассоциации и обобщения. Каждая ассоциация несет информацию о связях между объектами внутри ПС. Наиболее часто используются бинарные ассоциации, связывающие два

класса. Ассоциация может иметь название, которое должно выражать суть отображаемой связи (см. рис. 2). Помимо названия, ассоциация может иметь такую характеристику, как множественность. Она показывает, сколько объектов каждого класса может участвовать в ассоциации. Множественность указывается у каждого конца ассоциации (полюса) и задается конкретным числом или диапазоном чисел. Множественность, указанная в виде звездочки, предполагает любое количество (в том числе, и ноль).

Ассоциация «включает» показывает, что набор может включать несколько различных товаров. В данном случае направленная ассоциация позволяет найти все виды товаров, входящие в набор, но не дает ответа на вопрос, входит ли товар данного вида в какой-либо набор.

2.3 Проектирование базы данных

2.3.1 Логическая модель

Общим способом представления логической модели БД является построение ER-диаграмм (Entity-RElationship - сущность-связь). В этой модели сущность определяется как дискретный объект, для которого сохраняются элементы данных, а связь описывает отношение между двумя объектами.

Логический (концептуальный) уровень построен с учетом специфики и особенностей конкретной СУБД. Этот уровень представления данных ориентирован больше на компьютерную обработку и на программистов, которые занимаются ее разработкой. На этом уровне формируется концептуальная модель данных, то есть специальным способом структурированная модель предметной области, которая отвечает особенностям и ограничениям выбранной СУБД. [2]

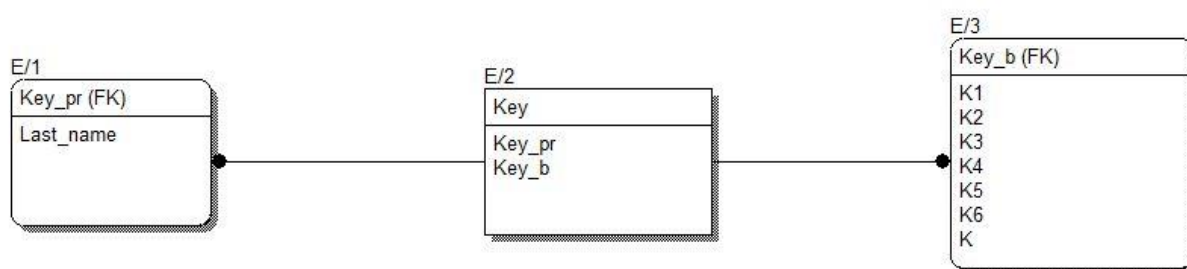


Рисунок 2.7 - Логический уровень

2.3.2 Физическая модель

Физическая модель данных зависит от конкретной СУБД, фактически являясь отображением системного каталога. В физической модели содержится информация обо всех объектах БД. Поскольку стандартов на объекты БД не существует (например, нет стандарта на типы данных),

физическая модель зависит от конкретной реализации СУБД. Следовательно, одной и той же логической модели могут соответствовать несколько разных физических моделей. Если в логической модели не имеет значения, какой конкретно тип данных имеет атрибут, то в физической модели важно описать всю информацию о конкретных физических объектах - таблицах, колонках, индексах, процедурах и т.д. Разделение модели данных на логические и физические позволяет решить несколько важных задач.

Связь многие-ко-многим возможна только на уровне логической модели данных, поэтому при переходе к физическому уровню ERWin автоматически преобразует связь многие-ко-многим, добавляя новую, ассоциативную сущность и устанавливая две новые связи один-ко-многим от старых к новой сущности [2].

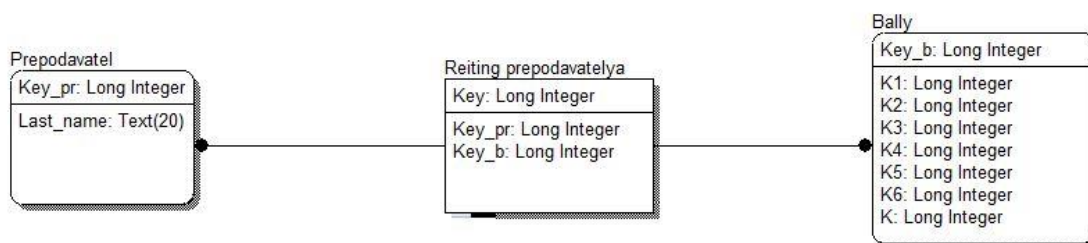


Рисунок 2.8 - Физический уровень

3 Программная реализация основных функциональных модулей системы

3.1 Разработка логики работы программ и пользовательских интерфейсов

Интерфейс любой системы является одной из очень важной составляющей. Он ориентирован, прежде всего, на конечного пользователя. От того, насколько хорошо спроектирован интерфейс зависят такие факторы как:

- а) скорость освоения (обучения пользователей) системы
- б) затраты на внедрение (обучится сам или специалисты помогут)
- в) последующая успешная работа (интерфейс должен быть понятен всегда, с ним всегда найдется возможность поработать), т.е. снижение риска возникновения ошибок пользователей от работы с системой

Сейчас ясно прослеживается тенденция отделения разработки пользовательского интерфейса от разработки остального приложения. Это связано с затратой большого количества времени именно на построение логики и формирование самого интерфейса.

Программное обеспечение для разработки пользовательского интерфейса разделяется на две основные группы - инструментарий для разработки пользовательского интерфейса (toolkits) и высокоуровневые средства разработки интерфейса (highEr-lEvEl dEvElopMEnt tools).

Инструментарий для разработки пользовательского интерфейса, как правило, включает в себя библиотеку примитивов компонентов интерфейса (меню, кнопки, полосы прокрутки и др.) и предназначен для использования программистами.

Высокоуровневые средства разработки интерфейса могут быть использованы непрограммистами и снабжены языком, который позволяет специфицировать функции ввода-вывода, а также определять, используя технику непосредственного манипулирования, интерфейсные элементы

Пользовательский интерфейс представляет несколько элементов управления. Каждый элемент может представлять следующую информацию:

- а) текстовую (текстовые поля ввода, метки, подписи)
- б) числовую (поля числового ввода, различные индикаторы прогресса, ползунки)
- в) графическую (изображения, индикаторы)
- г) звуковую (звуковые сообщения, команды)
- д) бинарную (флажки, индикаторы, радиокнопки)
- е) и другие

Реагировать элементы управления могут на действия пользователя, произведенных с помощью средств, таких как: мышь, клавиатура, сенсорный экран, голос, другие.

Одной из основных характеристик элемента системы является его состояние. И в каждом состоянии элементы могут отображаться и реагировать по-разному. Это необходимо, чтобы пользователь не мог привести систему в запрещенное состояние.

Логику пользовательского интерфейса представляют в объектно-ориентированном виде. Объект – это форма, у нее есть свойства и методы. Например, форма – регистрации. Свойства – поля, данные. Методы – кнопки, или действия, которые можно совершить.

Экранные формы должны проектироваться с учетом требований унификации:

- все экранные формы пользовательского интерфейса должны быть выполнены в едином графическом дизайне, с одинаковым расположением основных элементов управления и навигации;
- для обозначения сходных операций должны использоваться сходные графические значки, кнопки и другие управляющие (навигационные) элементы.

3.2 Разработка алгоритмов обработки запросов пользователей

Обработка запросов пользователей будет производиться через систему управления реляционными базами данных (СУРБД) - Microsoft SQL SErvEr 8.0. Основной используемый язык запросов — Transact-SQL, создан совместно Microsoft и SyBasE. Transact-SQL является реализацией стандарта ANSI/ISO по структурированному языку запросов (SQL) с расширениями. Используется для работы с базами данных размером от персональных до крупных баз данных масштаба предприятия; конкурирует с другими СУБД в этом сегменте рынка.

Microsoft SQL SErvEr - одна из наиболее мощных систем работы с базами данных в архитектуре "клиент-сервер". Особенность системы - работа сервера только в операционных системах ряда Microsoft Windows NT - NT SErvEr 4.0, 2000 SErvEr, SErvEr 2003, при этом клиентская часть может взаимодействовать с сервером из Microsoft Windows 98 и других операционных систем. Рекомендуемая файловая система для SQL SErvEr - NTFS, хотя возможна работа и в системе FAT [14].

В своем составе система имеет средства создания баз данных, работы с информацией баз данных, перенесения данных из других систем и в другие системы, резервного копирования и восстановления данных, развитую систему транзакций, систему репликации данных, реляционную подсистему для анализа, оптимизации и выполнения запросов клиентов, систему безопасности для управления правами доступа к объектам базы данных и пр.

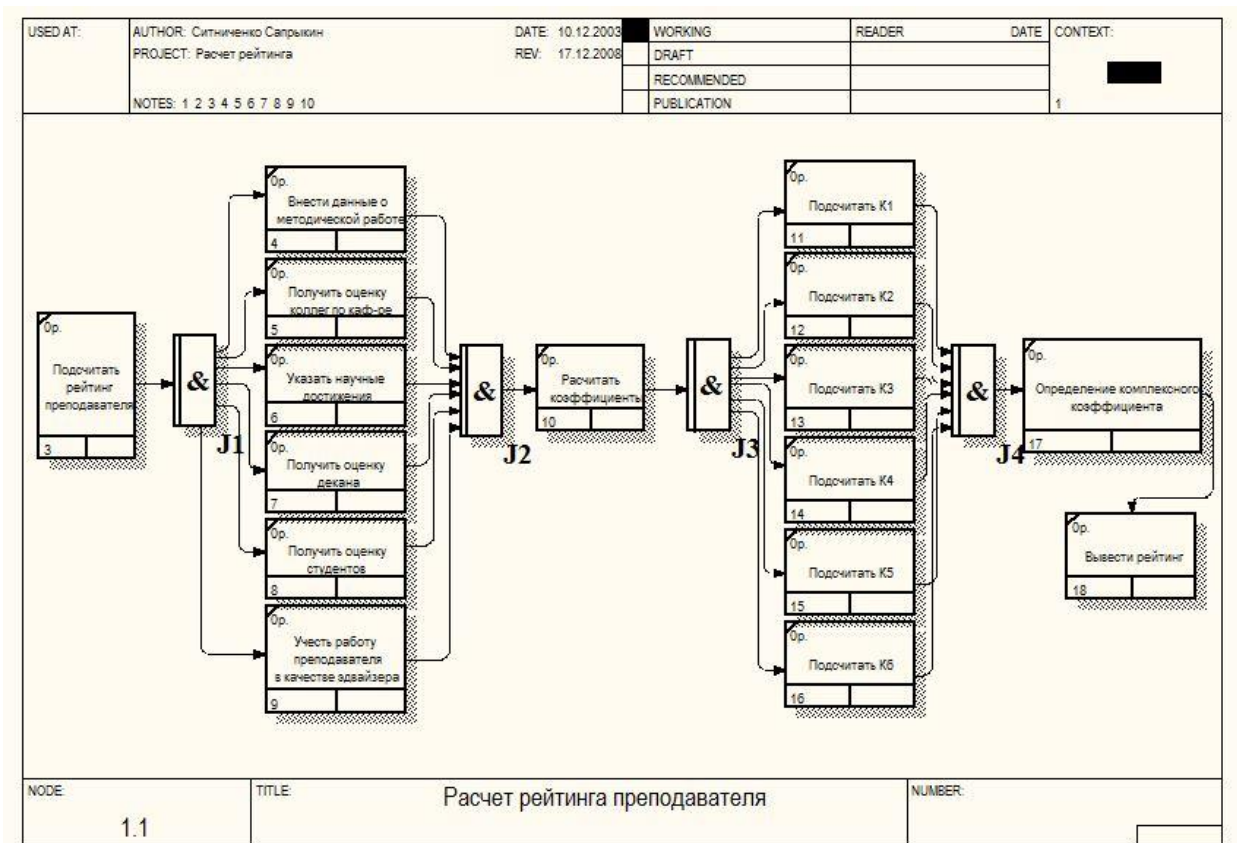


Рисунок 3.1 - Алгоритм обработки запросов пользователей

В данной работе тип связи был выбран: временное предшествование (TEmporal prEsEdEncE), простая стрелка. Исходное действие должно завершиться, прежде чем конечное действие сможет начаться.

Ветвление процесса отражается с помощью специального блока: "И", блок со знаком &.

В данной работе действия не должны выполняться синхронно, поэтому они обозначаются с одинарными вертикальными линиями внутри блока. [3]

3.3 Описание назначения и функций программ, режимов работы

АИС «Учет деятельности преподавателей кафедры «Информационные системы» предназначена для автоматизированного расчета рейтинга преподавателей и заведующего кафедрой ИС, в частности, для исполнения следующих процессов:

- хранение в базе данных информации о преподавателях и заведующим кафедрой;
- расчет коэффициентов для подсчета рейтинга;
- хранение и вывод информации о рейтинге преподавателей и заведующим кафедрой.

В составе АИС " Учет деятельности преподавателей кафедры «Информационные системы»должна решаться задача расчета рейтинга.

Подсистема расчета рейтинга предназначена для расчета рейтинга преподавателя с последующим начислением ему надбавки от администрации университета. Для подсчета рейтинга учитывается ряд коэффициентов, подсчет которых необходим для составления комплексного коэффициента. При расчете рейтинга преподавателя учитываются такие факторы, как результат работы преподавателя за последние 2 года, оценка труда преподавателя другими преподавателями кафедры, его научные достижения, оценка его работы деканом, оценка его работы студентами и оценка его труда в качестве эдвайзера группы.

При подсчете рейтинга заведующего кафедры учитываются практически те же факторы, что и для преподавателя, за исключением таких специфичных параметров, как оценка деятельности заведующего кафедрой.

Для АИС «Учет деятельности преподавателей кафедры «Информационные системы» определены следующие режимы функционирования:

- нормальный режим функционирования;
- аварийный режим функционирования.

Основным режимом функционирования АИС является нормальный режим.

В нормальном режиме функционирования системы:

– программное обеспечение и технические средства обеспечивают возможность круглосуточного функционирования, с перерывами на обслуживание;

- исправно работает оборудование;
- исправно функционирует программное обеспечение системы.

Для обеспечения нормального режима функционирования системы необходимо выполнять требования и выдерживать условия эксплуатации программного обеспечения и технических средств системы.

Аварийный режим функционирования системы характеризуется отказом одного или нескольких компонент программного и (или) технического обеспечения.

В случае перехода системы в аварийный режим необходимо:

- завершить работу всех приложений, с сохранением данных;
- выполнить резервное копирование БД.

После этого необходимо выполнить комплекс мероприятий по устранению причины перехода системы в аварийный режим.

3.4 Описание категорий пользователей программы, разграничения прав доступа и разделение привилегий

Категории пользователей программы АИС "Учет деятельности преподавателей кафедры «Информационные системы»:

- администратор баз данных;
- пользователь.

Основными обязанностями администратора баз данных являются:

- установка, модернизация, настройка параметров программного обеспечения СУБД;
- оптимизация прикладных баз данных по времени отклика, скорости доступа к данным;
- разработка, управление и реализация эффективной политики доступа к информации, хранящейся в прикладных базах данных.

Администратор баз данных должен обладать высоким уровнем квалификации и практическим опытом выполнения работ по установке, настройке и администрированию используемых в ИУС СУБД.

Пользователи системы должны иметь опыт работы с персональным компьютером на базе операционных систем Microsoft Windows на уровне квалифицированного пользователя и свободно осуществлять базовые операции в стандартных Windows.

Рекомендуемая численность для эксплуатации АИС «Учет деятельности преподавателей кафедры «Информационные системы»:

- администратор - 1 штатная единица;
- пользователь - число штатных единиц определяется структурой кафедры.

3.5 Реализованные части программы

В дипломном проекте реализованы все части программы, которые позволяют рассчитать рейтинг преподавателей и заведующего кафедры:

- авторизация пользователей;
- расчет коэффициентов К1 и К3 для преподавателей;
- просмотр для преподавателя его рейтинга;
- форма для расчета рейтинга преподавателей для секретаря;
- форма для расчета рейтинга для заведующего кафедры;
- форма для просмотра рейтингов всех преподавателей для заведующего кафедры;
- регистрация новых пользователей для секретаря.

3.6 Проектирование пользовательского интерфейса

Данная система содержит следующие окна:

- начальную форму;
- окно авторизации;
- окно «Результаты деятельности преподавателя»;
- окно «Рейтинг преподавателя»;
- окно «Рейтинги преподавателей»;
- окно «Рейтинг заведующего кафедры»;
- окно «Результат деятельности заведующего кафедры».

Начальная форма «Преподавательская». На этой форме расположен список преподавателей в алфавитном порядке.

The screenshot shows a window titled "Преподавательская" containing a table of teachers. The table has three columns: "Фамилия" (Surname), "Имя" (Name), and "Отчество" (Patronymic). The first row is highlighted with a mouse cursor. Below the table are three navigation buttons (left arrow, right arrow, plus sign) and two main buttons: "Обновить базу" (Refresh database) and "Выйти" (Exit).

Фамилия	Имя	Отчество
Абсатарова	Бибигуль	Рыскуловна
Абылхасенова	Дина	Кабылбековна
Адилгажинова	Сайран	Адилгажиновна
Адильбекова	Алия	Кудайбергеновна
Айткулов	Жалау	Сагинбаевич
Алибиева	Жибек	Мейрамбековна
Альмуратова	Камшат	Бимуратовна
Букашев	Аманжол	Абанович
Бычков	Андрей	Григорьевич
Водолазкина	Наталья	Александровна
Дьячков	Владимир	Васильевич
Жаксибаев	Мелдибек	Жумабекович
Зияханов	Мухтар	Умирзакович
Ибраев	Мажит	Сулейманович
Казахбаева	Гульжан	Ускенбаевна

Рисунок 3.1 – Основная форма

При двойном щелчке по фамилии преподавателя выходит форма авторизации.

Авторизация

Логин

Пароль

Преподаватель

Секретарь

Заведующий

Рисунок 3.3 – Авторизация

На этом окне можно выбрать свой уровень допуска к программе.

При вводе той фамилии преподавателя, по которой мы дважды щелкнули курсором мышки, и выборе уровня доступа Преподаватель, мы переходим на форму, где преподаватель может заполнить данные о своей методической и научной деятельности. Таким образом подсчитываются коэффициенты $K1$ и $K3$.

Результаты деятельности преподавателя

Методическая работа преподавателя **Научная работа преподавателя**

	Баллы за ед.	Количество	Баллы по количеству:
Учебник	80	<input type="text" value="0"/>	<input type="text" value="0"/>
Учебное пособие	60	<input type="text" value="0"/>	<input type="text" value="0"/>
Конспект лекций	40	<input type="text" value="0"/>	<input type="text" value="0"/>
Участие в разработке	40	<input type="text" value="0"/>	<input type="text" value="0"/>
Методическое указание	25	<input type="text" value="0"/>	<input type="text" value="0"/>
Физическая установка	80	<input type="text" value="0"/>	<input type="text" value="0"/>
Виртуальная работа	40	<input type="text" value="0"/>	<input type="text" value="0"/>
Не выполненные:			
Методическое указание	12	<input type="text" value="0"/>	<input type="text" value="0"/>
Учебное пособие	30	<input type="text" value="0"/>	<input type="text" value="0"/>

Итоговый балл:

Рисунок 3.4 – Результат деятельности преподавателя. Методическая работа преподавателя.

На этой форме преподаватель заполняет данные о своей методической работе. Программа сама рассчитывает итоговый балл по формуле расчета коэффициента K_1 .

Результаты деятельности преподавателя

Методическая работа преподавателя Научная работа преподавателя

Руководство темой:	60	0	0	Публикация монографии:	Издание РК:	60	0	0	Доклады на конференциях:	Республиканский ур-нь:	10	0	0
	35	0	0		Издание Заруб-е:	80	0	0		Международный ур-нь:	20	0	0
	25	0	0		Дальнее Заруб-е:	125	0	0		Дальнее зарубежье:	30	0	0
Исполнитель темы:				Получение и регистрация:	Диплом на открытие:	150	0	0		Работа в комиссиях:	40	0	0
Бюджетные:	30	0	0		Инов-ий патент в РК:	35	0	0	Конкурсные комиссии:	Факультетские:	5	0	0
Хоздоговорные:	20	0	0		Инов-ий патент в РФ:	60	0	0		Университетские:	15	0	0
Не финансируемые:	15	0	0		Инов-ий патент в мире:	70	0	0		Рук-о н-ым семинаром:	25	0	0
Докторская:	75	0	0	Публикация статьи:	РК не входящие в ККОН:	10	0	0		Руководство НИРС:	20	0	0
PhD:	60	0	0		В ближнем заруб-ье:	25	0	0		Рук-во призерами НИРС:	20	0	0
Кандидатская:	50	0	0		В дальнем заруб-ье:	30	0	0		Рук-во призи-унчих туров:	35	0	0
Руководство К:	50	0	0		За цитируемость в РК:	3	0	0		Рук-во призи-унчих туров:	45	0	0
Консультация док-й:	50	0	0		За цитируемость в мире:	8	0	0		Рук-во вып-ни бакалавров и призеров:	45	0	0
Консультация PhD:	50	0	0										

Итоговый балл:

Рисунок 3.5 - Результат деятельности преподавателя. Научная работа преподавателя.

В этом окне преподаватель вводит данные о своей научной деятельности. Программа так же сама рассчитывает и выводит итоговый балл.

После того, как данные подсчитаны, нужно нажать кнопку «Подсчитать». После этого появится поле «Введите итоговый балл», куда нужно ввести только что подсчитанный балл.

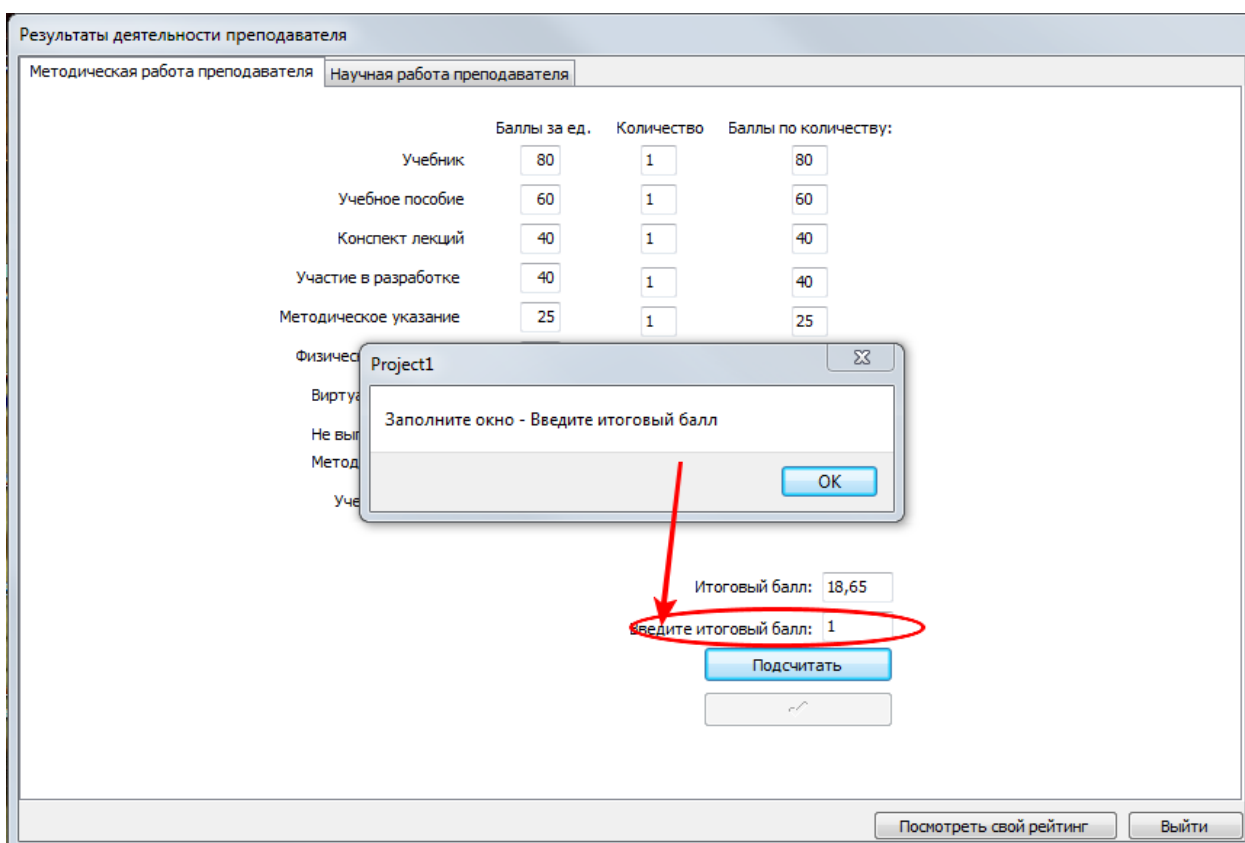


Рисунок 3.6 – Результат деятельности преподавателя. Методическая работа преподавателя. Ввод итогового балла.

В результате этого действия введенные баллы будут внесены в базу и их потом можно будет посмотреть.

При нажатии кнопки «Посмотреть свой рейтинг» преподаватель может ознакомиться со своим рейтингом.

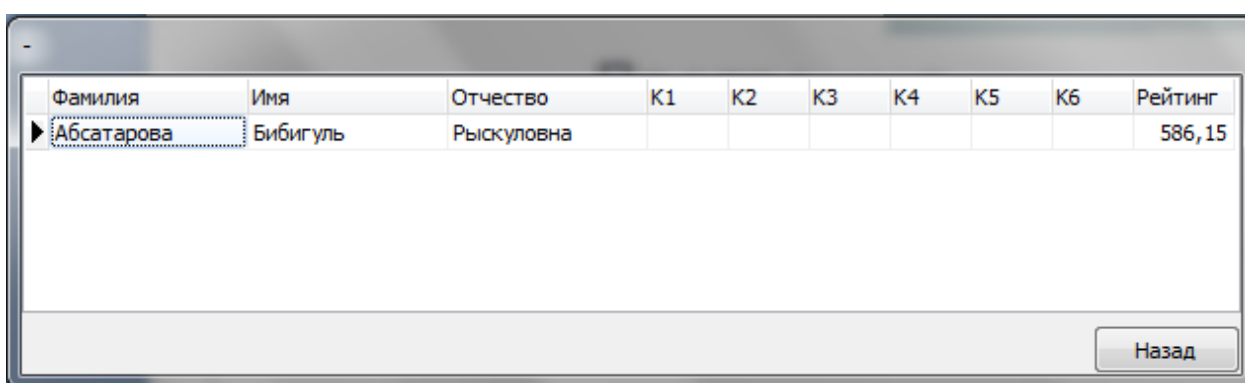


Рисунок 3.7 – Рейтинг

Как видно, данные о методической и научной работах внесены правильно.

На этом окне преподаватель так же может посмотреть и остальные показатели, оценивающие его работу.

Если на форме «Авторизация» в поле «Логин» ввести «Секретарь» и выбрать уровень доступа Секретарь, то открывается окно «Рейтинг преподавателя».

Фамилия	Имя	Отчество	K1	K2	K3	K4	K5	K6	Рейтинг
Абсатарова	Бибигуль	Рыскуловна	50,95	1	451,2	45	23	15	586,15

Методическая работа:

Оценка другими сотрудниками кафедры:

Научная деятельность:

Оценка труда с точки зрения декана:

Оценка труда преподавателя студентами:

Работа преподавателя в качестве эдвайзера:

Рейтинг преподавателя:

Рисунок 3.8 – Рейтинг преподавателя

В данном окне, секретарь может внести полученные данные по коэффициентам ДСК (оценка работы преподавателя другими преподавателями кафедры), ОПД (оценка работы преподавателя деканом), ОПС (оценка работы преподавателя студентами) и посчитать, а так же обновить рейтинг самого преподавателя в базе. Коэффициенты МР и НД секретарь менять не может, это предусмотрено безопасностью системы.

Если на форме «Авторизация» в поле «Логин» ввести фамилию заведующего кафедры и выбрать уровень доступа Заведующий, то выходит форма «Рейтинги преподавателей».

Фамилия	Имя	Отчество	K1	K2	K3	K4	K5	K6	Рейтинг
▶ Бычков	Андрей	Григорьевич	0	0	0	0	0	0	0
Рахимбергенов	Султан	Рахимбергенович	0	0	0	0	0	0	0
Ибраев	Мажит	Сулейманович	0	0	0	0	0	0	0
Казахбаева	Гульжан	Ускенбаевна	0	0	0	0	0	0	0
Курманкожаева	Айкын	Азимхановна	0	0	0	0	0	0	0
Дьячков	Владимир	Васильевич	0	0	0	0	0	0	0
Жаксимаев	Мелдибек	Жумабекович	0	0	0	0	0	0	0
Абылхасенова	Дина	Кабылбековна	10,25	0	6,25	0	0	0	0
Смагулова	Сауле	Ермековна	0	0	0	0	0	0	0
Водолазкина	Наталья	Александровна	0	0	0	0	0	0	0
Адилгажинова	Сайран	Адилгажиновна	0	0	0	0	0	0	0
Альмуратова	Камшат	Бимуратовна	0	0	0	0	0	0	0
Купарова	Айжан	Токумаевна	0	0	0	0	0	0	0
Каирбаева	Балнур	Каирбаевна	0	0	0	0	0	0	0
Калдышева	Индира	Нурабиловна	0	0	0	0	0	0	0

Рисунок 3.9 – Рейтинги преподавателей

Эту форму может просматривать только заведующий кафедры для оценки работы всех преподавателей. Данная таблица сортируется по рейтингу, и по ней можно определить, на каком месте находится тот или иной преподаватель. Это необходимо для начисления надбавки преподавателю.

При нажатии кнопки «Подсчитать рейтинг» выходит форма «Рейтинг заведующего кафедры», где заведующий вводит различные показатели для расчета своего рейтинга.

Рейтинг заведующего кафедры

Фамилия Имя Отчество **Имангалиев Шнар Ихсанович**

К1: выпуск методическим разработок кафедры

К2: оценка работы заведующего преподавателями кафедры

К3: оценка работы преподавателей кафедры в области науки

К4: оценка испол. дисциплины и организ. работы заведующего

УМО	Проректор по НР и МС	<input type="text"/>	<input type="text"/>
Декан	Проректор по УМР	<input type="text"/>	<input type="text"/>

К5: оценка труда заведующего в качестве преподавателя

К6: оценка трудовой дисциплины сотрудников кафедры

К7: оценка работы преподавателей кафедры по внедрению новой техники и модернизации лабораторного оборудования

К8: оценка методической работы заведующего кафедры

Итоговый балл:

Введите итоговый балл:

Рисунок 3.10 - Рейтинг заведующего кафедры

Для подсчета оценки методической работы заведующего кафедры нужно нажать на кнопку «Расчет К8». После этого открывается форма «Результаты деятельности заведующего кафедры».

Методическая работа	25,55
Научная работа	117,7
Итоговый балл:	

Расчитать

Рисунок 3.11 - Результаты деятельности заведующего кафедры

На этой форме заведующий вводит данные о своей научной и методической деятельности, а затем подсчитывает коэффициент К8, который равен сумме двух оценок, разделенных на 40.

4 Техничко-экономическое обоснование

Автоматизированная система, разрабатываемая в рамках дипломной работы, предназначена для облегчения подсчета рейтинга преподавателей и заведующего кафедры Информационных систем.

Назначение и цели создания (развития) системы:

- назначение системы – проектируемая система предназначена для использования её сотрудниками кафедры: заведующим кафедры, преподавателями и секретарем. Её назначение – расчет рейтинга преподавателей кафедры;

- цели создания системы – снижение трудоемкости обработки, создания, хранения и передачи информации на 70% за счет создания систематизированной базы данных, облегчения подсчета рейтинга и нагрузки за счет автоматизации этих процессов.

АИС «Учет деятельности преподавателей кафедры «Информационные системы» предназначена для автоматизированного расчета рейтинга преподавателей кафедры ИС, в частности, для исполнения следующих процессов:

- сбор данных для расчета рейтинга;
- хранение в базе данных информации о преподавателях;
- расчет коэффициентов для подсчета рейтинга;
- вывод информации о рейтинге преподавателя по семестрам и за год;
- хранение и вывод информации о рейтинге преподавателей.

Основными целями создания АИС «Учет деятельности преподавателей кафедры «Информационные системы» являются:

- сбор информации для расчета;
- автоматизирование расчета рейтинга преподавателя;
- повышение эффективности исполнения расчетов рейтинга путем сокращения непроизводительных и дублирующих операций, операций, выполняемых "вручную", оптимизации информационного взаимодействия участников процессов;
- исключение ошибок при расчетах;
- повышение качества принятия управленческих решений за счет оперативности представления, полноты, достоверности и удобства форматов отображения информации;
- обеспечение целостности, доступности и конфиденциальности информации.

Пользователями системы являются все сотрудники кафедры Информационных систем.

Из выше сказанного можно сделать вывод, что разрабатываемый ПП значительно облегчит подсчет такого важного показателя оценки работы

преподавателя как рейтинг, а также практически исключит ошибки, получаемые под воздействием человеческого фактора.

Ниже будет проведен ряд вычислений для определения экономической эффективности от внедрения разрабатываемой программы.

После этого можно будет судить о целесообразности внедрения данного программного продукта.

4.1 Расчет трудоемкости разработки Автоматизированной системы

Для определения трудоемкости разработки ПП составляется перечень всех основных этапов и видов работ, которые были выполнены. При этом уделяется особое внимание порядку выполнения отдельных видов работ, так необходимо соблюдать логическую последовательность для выявления возможностей параллельного выполнения некоторых пунктов. Это позволит сократить время проведения разработки ПП [13].

Форма разделения работ по этапам с указанием трудоемкости их выполнения приведена в таблице 4.1.

Таблица 4.1 - Распределение работ по этапам и видам и оценка их трудоемкости

№ этапа	Содержание этапа	Время, затраченное на данный этап, час.
1	Описание требуемой задачи	10
2	Разработка технического задания	10
3	Разработка моделей задачи	20
4	Создание базы данных	60
5	Написание кода программы	80
6	Компоновка всех частей программного продукта	50
7	Отладка и тестирование программы.	40
8	Оформление документации, инструкции пользователю, пояснительной записки	40
		310

Общее количество часов активной работы по разработке программного продукта равно 310. В сутки на разработку выделялось 7 часов, следовательно, срок выполнения проекта равен 44 суткам. Для дальнейших расчетов время разработки программного продукта округляем до двух месяцев.

4.2 Расчет затрат на разработку АИС

Основная заработная плата за разработку ПП. Основная заработная плата за разработку ПП определяется в зависимости от тарифной ставки разработчика за час работы и времени затраченного на каждый этап работы.

Общая сумма затрат на оплату труда ($Z_{тр}$) определяется по формуле:

$$Z_{тр} = \sum ЧС_i \times T_i, \quad (4.1)$$

где $ЧС_i$ - часовая ставка i -го работника, тг;

T_i - трудоемкость разработки ПП, чел.×ч;

i - категория работника;

n - количество работников, занятых разработкой ПП [13].

Таблица 4.2- Расчёт основной заработной платы разработчиков ПП

Наименование и содержание этапов работ	Исполнитель	Трудоемкость, норма-час	Заработная плата за час работы	Сумма заработной платы, тг
Описание требуемой задачи.	Руководитель проекта	10	2 000	20 000
Разработка технического задания	Руководитель проекта	10	2 000	20 000
Разработка моделей задачи	Менеджер по проектированию	20	1 500	3 000
Создание базы данных	Программист	60	1 200	72 000
Написание кода программы	Программист	80	1 200	96 000
Компоновка всех частей программного продукта	Программист	50	1 200	60 000
Отладка и тестирование программы.	Программист-отладчик	40	1 200	48 000
Оформление документации, инструкции пользователю, пояснительной записки	Руководитель проекта	40	2 000	80 000
Итого		310	12,3	399 000

Социальный налог – согласно Налоговому кодексу РК он составляет 11 % от ФОТ. Пенсионные отчисления не облагаются социальным налогом.

$$\text{Ос.} = (\text{ФОТ} - \text{ПО}) * 0,11, \quad (4.2)$$

где ПО - отчисления в пенсионный фонд, 10% от ФОТ.

$$\text{ПО} = \text{ФОТ} * 10\% = 399\,000_{\text{тг.}} * 0,1 = 39\,900_{\text{тг.}}$$

$$\text{Ос.} = (399\,000_{\text{тг.}} - 39\,900) * 0,11 = 39\,501 \text{ тг.}$$

Расчет расхода материалов и комплектующих
Общая сумма затрат на материальные ресурсы (Z_M) определяется по формуле:

$$Z_M = \sum P_i \times C_i, \quad (4.3)$$

где P_i - расход i -го вида материального ресурса, натуральные единицы;

C_i - цена за единицу i -го вида материального ресурса, тг;

i - вид материального ресурса;

n - количество видов материальных ресурсов.

Расчет затрат на материальные ресурсы производится по форме, приведенной в таблице 4.3.

Таблица 4.3 – Расчет расхода материалов и комплектующих

Наименование материала	Марка	Единицы измерения	Количество	Цена за единицу в тенге	Сумма в тенге
Мышка	HP	Штука	8	1500	12000
Клавиатура	GENIUS	Штука	3	2300	6900
Flash-носитель	TranscEnd	Штука	7	3000	21000
Компакт-диск (чистый)	LASER CRAFT	Упаковка	1	2000	200
Бумага	"Берер"	Упаковка	2	1000	2000
Картридж для принтера	DEskjEt 5150	Штука	1	3000	3000
Ручка	"PapEr MatE"	Упаковка	1	3000	3000
Блокнот	ABDY	Штука	10	1000	10000
Итого			33	16800	58100

Общая сумма затрат на электроэнергию ($Z_Э$) рассчитывается по формуле:

$$Z_Э = \sum M_i \times T_i \times C, \quad (4.4)$$

где M_i - паспортная мощность i -го электрооборудования, кВт;
 K_i - коэффициент использования мощности i -го электрооборудования (принят $K_i=0.7$);
 T_i - время работы i -го оборудования за весь период разработки ПП ч;
 Π - цена электроэнергии, тг/кВт×ч;
 i - вид электрооборудования;
 n - количество электрооборудования.

Затраты на электроэнергию приведены в таблице 4.3.

Таблица 4.4 - Затраты на электроэнергию

Наименование оборудования	Паспортная мощность, кВт	Коэфф-т использования мощности	Время работы оборуд-я для разработки ПП, ч	Цена электро энергии, $\frac{\text{тг.}}{\text{кВт} \times \text{ч}}$	Сумма, тг
ПК	0,4	0,7	230	22	1416,8
Ноутбук	0,2	0,9	60	22	237,6
МФУ	0,07	0,8	310	22	381,92
ИТОГО затраты на электроэнергию					2036,32

Расчет амортизационных отчислений (за время разработки ПП)

Сумма годовых амортизационных отчислений определяется по формуле:

$$A = \text{Перв. стоимость} * \text{Норма амортизации}/100, \quad (4.5)$$

Амортизационные отчисления приведены в таблице 4.5.

Таблица 4.5 - Расчет амортизационных отчислений для аппаратного обеспечения

Наименование оборудования	Первоначальная стоимость, тг	Норма амортизации	Сумма амортизации, тг	Сумма амортизации в месяц, тг.
Аппаратное обеспечение:				
Ноутбук ASUS N76VB	150 000	20	30 000	2500
Системный блок HP	120 000	20	24000	2000

Продолжение таблицы 4.5

Монитор Samsung	18 000	20	3600	300
Многофункциональное устройство Xerox Phaser	56000	20	11200	933,3333333
Итого			68800	5733,3

Таблица 4.6 - Расчет амортизационных отчислений для программного обеспечения

Наименование оборудования	Первоначальная стоимость, тг	Норма амортизации	Сумма амортизации, тг	Сумма амортизации в месяц, тг.
Программное обеспечение:				
Windows 7 Professional	12000	15	1800	150
EmBarcadEro RAD Studio7.0	8000	15	1200	100
Microsoft Office Home and Student 2010	3500	15	525	43,75
Итого			3525	293,75
Итого по всем амортизационным отчислениям				6027,05

Норма амортизации=100/СПИ(срок полезного использования)

$A = \text{Перв. стоимость} * \text{Норма амортизации} / 100$

$A_{мес} = \text{Сумма амортизации} / 12$

Сумма амортизационных отчислений за два месяца равна 12054,1 тг.

В статью «Прочие затраты» включаются расходы на арендную плату, включая коммунальные платежи, затраты на лицензирование и сертификацию, расходы на рекламу, канцелярские и прочие хозяйственные расходы.

Арендная плата. Арендная плата рассчитывается в зависимости от стоимости аренды 1 кв.м площади и времени разработки ПП. Постройте диаграмму структуры себестоимости ПП.

Офис организации, разрабатывающей программный продукт, расположен по адресу Фурманова, 88. Площадь арендуемого помещения составляет 100 кв.м., поскольку помимо основного и вспомогательного персонала есть непромышленный персонал. Стоимость аренды помещения в данном районе расположения офиса составляет 50 000 тг. в месяц, учитывая коммунальные услуги и интернет.

Арендная плата - денежная сумма, выплачиваемая арендатором за пользование арендуемым имуществом. Размер арендной платы определяется соглашением арендатора и арендодателя.

Арендная плата рассчитывается по формуле:

$$AP = Ca * S, \quad (4.6)$$

где Ca – срок аренды;

S – стоимость аренды за 1 месяц.

Арендная плата составляет:

$$AP = 2 * 50\,000 = 100\,000 \text{ тг.}$$

Расходы на Интернет. Поскольку общее время на разработку ПП 44 суток или округленно 2 месяца, то расходы на Интернет, месячная оплата которого составляет 5000 тг. равны:

$$P_{и} = 2 * 5000 = 10\,000 \text{ тг.}$$

Прочие хозяйственные расходы составляют 3 000 тг.

Прочие затраты = 100 000 + 10 000 + 3 000 = 113 000 тг.

На основании полученных данных по отдельным статьям в таблице 4.6 приведена смета затрат на разработку ПП

Таблица 4.7 – Структура себестоимости ПП

Статья затрат	Сумма, тг
Заработная плата основных разработчиков	399 000
Социальный налог	39 501
Амортизационные отчисления	12054,1
Расходные материалы	58100
Затраты на электроэнергию	2036,32
Прочие затраты	113 000

Продолжение таблицы 4.7

ИТОГО	623 691,42
-------	------------

4.3 Определение возможной (договорной) цены ПП

Величина возможной (договорной) цены ПП должна устанавливаться с учетом эффективности, качества и сроков ее выполнения на уровне, отвечающем экономическим интересам заказчика (потребителя) и исполнителя.

Договорная цена ($Ц_d$) для прикладных ПП рассчитывается по формуле:

$$Ц_d = Z_{\text{нир}} * \left(1 + \frac{P}{100}\right), \quad (4.7)$$

где $Z_{\text{нир}}$ - затраты на разработку ПП (из таблицы 4.6), тг;

P - средний уровень рентабельности ПП. % (принято 25%).

$$Ц_d = 623\,691,42 * (1 + 0,25) = 779\,614,27 \text{ тг.}$$

Цена реализации с учетом НДС рассчитывается по формуле:

$$Ц_p = Ц_d + Ц_d * \text{НДС.}$$

НДС, согласно Налоговому кодексу РК, составляет 12 %.

$$Ц_p = 779\,614,27 + 779\,614,27 * 0,12 = 873\,167,98 \text{ тг.}$$

Разработанный программный продукт будет лицензироваться корпоративным лицензированием, что позволяет лицензировать практически любой программный продукт для любого количества ПК в рамках одного или нескольких соглашений по выбору клиента. Этот вид лицензирования выбирается потому, что программа предусмотрена для работы на 3х компьютерах кафедры Информационных систем.

4.4 Оценка социально - экономических результатов функционирования автоматизированной системы

Созданная в рамках дипломного проекта автоматизированная система предназначена для расчета важного показателя оценки преподавателя – рейтинг. Этот показатель влияет на рост преподавателя по карьерной

лестнице, учитывается при выплате премий и надбавок к зарплате. Поэтому, очень важно, чтобы рейтинг был рассчитан правильно, чтобы при его расчете были сведены к минимуму ошибки, допущенные под воздействием человеческого фактора.

В современном мире темпы роста продуктов в сфере информационных технологий имеют колоссальную скорость. И для улучшения деятельности университета очень важно, чтобы его информационные ресурсы находились на должном уровне. Для этого и важны разработки таких автоматизированных систем, наличие которых показывают, что университет идет в ногу со временем.

Данная программа разрабатывалась для Алматинского Университета Энергетики и Связи, учитывались специфические для ВУЗа факторы для расчета рейтинга преподавателя: для преподавателя это 6 показателей, охватывающих все области его преподавательской деятельности. Для заведующего кафедры это 8 показателей, которые тоже охватывают все области его преподавательской, научной, руководящей деятельности. Но при разработке и программировании учитывалось то, что факторы расчета рейтинга могут измениться со временем. А так же, что этой программой могут заинтересоваться другие ВУЗы. Если такая ситуация наступит, то программист всегда сможет дополнить программу под новые условия.

На мой взгляд, программа является актуальной разработкой, поскольку аналогов ее в АУЭС не существует. Хотя она и разрабатывалась для кафедры Информационных систем, но в ее основе лежит документ «Методика оценки качества работы (определение рейтинга) профессорско-преподавательского состава некоммерческого АО «АУЭС». В программе предусмотрена авторизация, следовательно, ее можно будет применить для любой кафедры АУЭС.

5 Безопасность жизнедеятельности

5.1 Анализ опасных и вредных производственных факторов

Создание дипломной работы «Разработка и проектирование Автоматизированной информационной системы «Учет деятельности преподавателей кафедры «Информационные системы» проводится в помещении (кабинете) без повышенной серьезности, касающиеся специальной категории. Кабинет сухой, температура в нем поддерживалась на уровне 20°C – 25°C с влажностью в пределах 60%. Условия в кабинете соответствуют ГОСТу: нет сырости и токопроводящей пыли.

Данная работа представляет собой программу для сотрудников кафедры Информационных систем. Поэтому, для работы с этой программой, требуется персональный компьютер.

Кафедра состоит из двух комнат.

5.1.1 Анализ помещения:

Рабочее помещение находится на третьем этаже.

Тип аудитории: кафедра.

Помещение представляет собой две соединенные комнаты, размеры которых 5x3x3 и 5x6x3 (ширина-длина-высота)

Источники света: светильники – 3 шт. в одном кабинете и 4 шт. в другом. В каждом по 4 люминесцентные лампы.

Окон в помещении три (1+2).

Окраска кабинета и мебели по стандарту должна способствовать созданию благоприятных условий для зрительного восприятия.

Для защиты от избыточной яркости с окон могут быть применены жалюзи.

В помещении сидят 3 человека. Работают в дневную смену. Режим работы с 8:20 до 18:00.

Схема помещения приведена на рисунке 5.1.

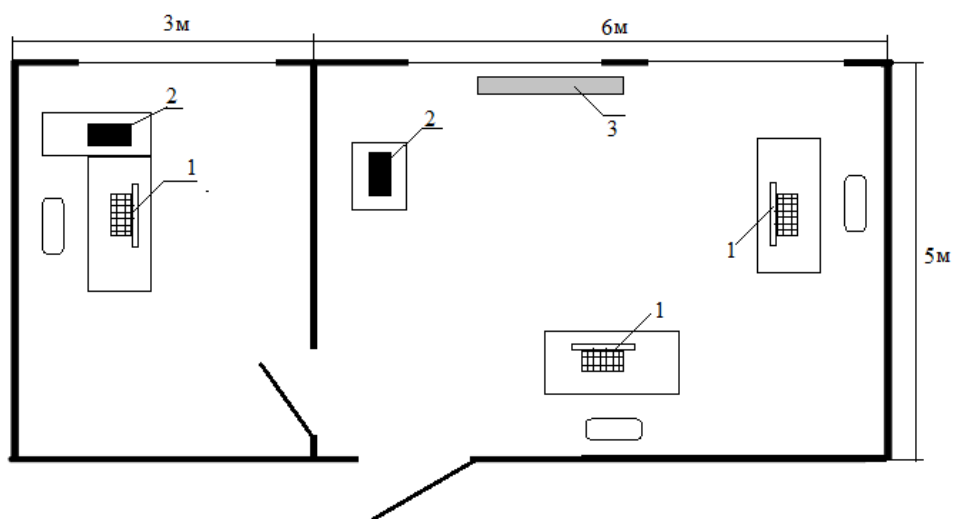


Рисунок 5.1– План помещения: 1 – ПК, 2 – принтер/сканер, 3 – кондиционер.

5.1.2 Характеристика оборудования

В таблице представлены характеристики используемого оборудования.

Таблица 5.1 – Оборудование и его характеристики

Наименование оборудования	Характеристики
Стационарный компьютер	Windows 7, Intel® Atom (TM) CPU D525 @1.80 GHz, 2048МБ, 320ГБ, Intel GMA HD, DOS, 15.6", DVD±RW (DL), VGA, HDMI, RJ-45, Super Multi Dual Layer, Intel HD Graphics) Количество – 3 шт., Мощность – 40Вт;
Название оборудования	Характеристики
Принтеры, сканеры или множительные аппараты	МФУ Canon MF 4430 Adf “3d 1”, i-sEnsys MF 4340d Количество 2-шт.
Кондиционер настенный GrEE -09 GW	Мощность по холоду 16.7 кВт Мощность компрессора 4.5 кВт Мощность электронагревателя 6.6 кВт

Воспроизведение визуальной информации на дисплее, которая должна вовремя и точно восприниматься пользователем, называется работой с персональным компьютером.

Главным фактором, влияющим на производительность труда пользователей, работающих с компьютером, являются комфортные и безопасные условия труда.

Условия труда пользователя, работающего с персональным компьютером, обуславливаются:

- особенностями организации рабочего места;
- обстоятельствами производственной среды (освещением, микроклиматом, шумом, электромагнитными и электростатическими полями, визуальными эргономическими параметрами дисплея и т. д.);
- характеристиками информационного взаимодействия человека и персональных электронно-вычислительных машин.

При выполнении работ на персональном компьютере (ПК) могут иметь место следующие факторы:

- повышенная температура поверхностей ПК;
- повышенная или пониженная температура воздуха рабочей зоны;
- выделение в воздух рабочей зоны ряда химических веществ;
- повышенная или пониженная влажность воздуха;
- повышенный или пониженный уровень отрицательных и положительных аэроионов;
- повышенное значение напряжения в электрической цепи, замыкание;
- повышенный уровень статического электричества;
- повышенный уровень электромагнитных излучений;
- повышенная напряженность электрического поля;
- отсутствие или недостаток естественного света;
- недостаточная искусственная освещенность рабочей зоны;
- повышенная яркость света;
- повышенная контрастность;
- прямая и отраженная блескость;
- зрительное напряжение;
- монотонность трудового процесса;
- нервно-эмоциональные перегрузки.

Увеличенные статические и динамические нагрузки у пользователей ПК приводят к жалобам на боли в спине, шейном отделе позвоночника и руках. Из всех недомоганий, обусловленных работой на компьютерах, чаще встречаются те, которые связаны с использованием клавиатуры. В период выполнения операций ввода данных количество мелких стереотипных

движений кистей и пальцев рук за смену может превысить 60 тыс., что в соответствии с гигиенической классификацией труда относится к категории вредных и опасных. Поскольку каждое нажатие на клавишу сопряжено с сокращением мышц, сухожилия непрерывно скользят вдоль костей и соприкасаются с тканями, вследствие чего могут развиваться болезненные воспалительные процессы.

К числу факторов, ухудшающих состояние здоровья пользователей компьютерной техники, относятся электромагнитное и электростатическое поля, акустический шум, изменение ионного состава воздуха и параметров микроклимата в помещении. Значительную роль играют эргономические параметры расположения экрана монитора (дисплея), состояние освещенности на рабочем месте, характеристики мебели и параметры помещения, где расположена компьютерная техника.

К физическим вредоносным и тяжелым факторам причисляются: повышенные уровни электромагнитного, рентгеновского, ультрафиолетового и инфракрасного излучения; повышенный уровень статического электричества и запыленности воздуха рабочей области; увеличенное содержание положительных аэронов и уменьшенное содержание отрицательных аэроионов в воздухе рабочей области; повышенный уровень блескости и ослепленности; неравномерность распределения яркости в поле зрения; увеличенная яркость светового изображения; повышенное значение напряжения в электрической цепи, замыкание которой может произойти через тело человека.

Химические вредные и опасные факторы следующие: повышенное содержание в воздухе рабочей зоны двуокси углерода, озона, аммиака, фенола и формальдегида.

Психофизиологические вредные и опасные факторы: напряжение зрения и внимания; интеллектуальные, эмоциональные и длительные статические нагрузки; монотонность труда; большой объем информации, обрабатываемый в единицу времени; нерациональная организация рабочего места.

Тепловое воздействие ПК характеризуется повышением температуры тела, локальным избирательным нагревом клеток, тканей и органов вследствие перехода ПК в тепловую энергию. Интенсивность нагрева зависит от количества поглощенной энергии и скорости оттока тепла от облучаемых участков тела. Отток тепла затруднен в органах и тканях с плохим кровоснабжением. К ним в первую очередь относится хрусталик глаза, вследствие чего возможно развитие катаракты. Тепловому воздействию ЭМП подвергаются также паренхиматозные органы (печень, поджелудочная железа) и полые органы, содержащие жидкость (мочевой пузырь, желудок). Нагревание их может вызвать обострение хронических заболеваний [19].

5.2 Защитные мероприятия

5.2.1 Мероприятия, обеспечивающие безопасность при работе на компьютере

Организация рабочего места в соответствии с требованиями эргономики.

Под рабочим местом условно понимают зону, оснащенную необходимыми техническими средствами, где работник или группа работников постоянно или временно выполняют одну работу или операцию.

Правильная планировка рабочих мест может существенно снизить действие целого ряда неблагоприятных факторов, действующих на работающего с ПЭВМ человека.

Санитарными правилами предусмотрены следующие нормы.

Помещения с ВДТ (видео-дисплейные терминалы) и ПЭВМ должны иметь естественное и искусственное освещение.

Естественное освещение должно осуществляться через светопроемы, ориентированные преимущественно на север и северо-восток и обеспечивать коэффициент естественной освещенности (КЕО) не ниже 1.2.

Для внутренней отделки интерьера помещений с ВДТ и ПЭВМ должны использоваться диффузно-отражающие материалы с коэффициентом отражения для потолка – 0.7 – 0.8; для стен – 0.5 – 0.6; для пола – 0.3 – 0.5.

Конструкция рабочего стула (кресла) должна обеспечивать поддержание рациональной рабочей позы при работе на ВДТ и ПЭВМ, позволять изменять позу с целью снижения статического напряжения мышц шейно-плечевой области и спины для предупреждения развития утомления.

Рабочий стул (кресло) должен быть подъемно-поворотным и регулируемым по высоте и углам наклона сиденья и спинки, а также расстоянию спинки от переднего края сиденья, при этом регулировка каждого параметра должна быть независимой, легко осуществляемой и иметь надежную фиксацию.

Экран видеомонитора должен находиться от глаз пользователя на оптимальном расстоянии 600 – 700 мм, но не ближе 500.

Рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной – не менее 500 мм, глубиной на уровне колен – не менее 450 мм и на уровне вытянутых ног – не менее 650 мм.

Клавиатуру следует располагать на поверхности стола на расстоянии 100 – 300 мм от края, обращенного к пользователю или на специальной, регулируемой по высоте рабочей поверхности, отделенной от основной столешницы.

Рациональная организация режима труда и отдыха. Для предупреждения заболеваний, связанных с работой на компьютере необходима рациональная организация труда и отдыха, которая нормируется в соответствии с санитарными правилами.

Режимы труда и отдыха при работе с ПК должны организовываться в зависимости от вида и категории трудовой деятельности.

Для обеспечения оптимальной работоспособности, сохранения здоровья пользователей, а также предупреждения развития переутомления на протяжении рабочей смены (учебных занятий) должны устанавливаться регламентированные перерывы.

Для студентов ВУЗов длительность работ за ПК во время учебных занятий определяется курсом обучения, характером (ввод данных, программирование, отладка программ, редактирование и др.) и сложностью выполняемых заданий, а также техническими данными ПЭВМ и их разрешающей способностью.

5.2.2 Электробезопасность

На рабочем месте пользователя расположены дисплей, клавиатура и системный блок. При включении дисплея на электронно-лучевой трубке создается высокое напряжение в несколько киловольт. Вследствие этого воспрещается прикасаться к тыльной стороне дисплея, вытирать пыль с компьютера при его включенном состоянии, работать на компьютере во влажной одежде и влажными руками.

Перед началом работы следует убедиться в отсутствии свешивающихся со стола или висящих под столом проводов электропитания, в целостности вилки и провода электропитания, в отсутствии видимых повреждений аппаратуры и рабочей мебели, в отсутствии повреждений и наличии заземления приэкранного фильтра.

Токи статического электричества, наведенные в процессе работы компьютера на корпусах монитора, системного блока и клавиатуры, могут приводить к разрядам при прикосновении к этим элементам. Такие разряды опасности для человека не представляют, но могут привести к выходу из строя компьютера. Для снижения величин токов статического электричества используются нейтрализаторы, местное и общее увлажнение воздуха, использование покрытия полов с антистатической пропиткой.

5.2.3 Пожарная безопасность

Пожарная безопасность — состояние объекта, при котором исключается возможность пожара, а в случае его возникновения предотвращается воздействие на людей опасных его факторов и обеспечивается защита материальных ценностей.

Противопожарная защита — это комплекс организационных и технических мероприятий, направленных на обеспечение безопасности людей, предотвращение пожара, ограничение его распространения, а также на создание условий для успешного тушения пожара.

Пожарная безопасность обеспечивается системой предотвращения пожара и системой пожарной защиты. Во всех служебных помещениях обязательно должен быть «План эвакуации людей при пожаре», регламентирующий действия персонала в случае возникновения очага возгорания и указывающий места расположения пожарной техники.

Пожары на кафедрах представляют особую опасность, так как сопряжены с большими материальными потерями. Характерная особенность кафедры — небольшие площади помещений. Как известно, пожар может возникнуть при взаимодействии горючих веществ, окислителя и источников зажигания. В помещениях кафедры присутствуют все три основных фактора, необходимые для возникновения пожара.

Источниками зажигания на кафедре могут быть электрические схемы от ЭВМ, приборы, применяемые для технического обслуживания, устройства электропитания, кондиционирования воздуха, где в результате различных нарушений образуются перегретые элементы, электрические искры и дуги, способные вызвать загорания горючих материалов.

В современных ЭВМ очень высокая плотность размещения элементов электронных схем. В непосредственной близости друг от друга располагаются соединительные провода, кабели. При протекании по ним электрического тока выделяется значительное количество теплоты. При этом возможно оплавление изоляции. Для отвода избыточной теплоты от ЭВМ служат системы вентиляции и кондиционирования воздуха. При постоянном действии эти системы представляют собой дополнительную пожарную опасность.

Для большинства помещений, содержащих ПК, установлена категория пожарной опасности В.

Одна из наиболее важных задач пожарной защиты — защита строительных помещений от разрушений и обеспечение их достаточной прочности в условиях воздействия высоких температур при пожаре. Учитывая высокую стоимость электронного оборудования ВЦ, а также категорию его пожарной опасности, здания для ВЦ и части здания другого назначения, в которых предусмотрено размещение ЭВМ, должны быть первой и второй степени огнестойкости. Для изготовления строительных конструкций используются, как правило, кирпич, железобетон, стекло, металл и другие негорючие материалы. Применение дерева должно быть ограничено, а в случае использования необходимо пропитывать его огнезащитными составами.

5.2.4 Аналитический расчет искусственного освещения

Освещенность рабочей поверхности, создаваемая светильниками общего освещения в системе комбинированного, должна составлять не менее 10% нормируемой для комбинированного освещения при тех источниках

света, которые применяются для местного освещения. При этом освещенность должна быть не менее 400 лк при люминесцентных лампах. В таблице 5.2 приведены нормы освещенности при искусственном освещении.

Таблица 5.2 - Нормы освещенности при искусственном освещении

Характеристика зрительной работы	Наименьший или эквивалентный размер объекта различения, мм.	Разряд зрительной работы	Искусственном освещении
			Освещенность, лк
			при комбинированном освещении.
Средней точности	Свыше 0.5 до 1.0	IV	400

Расчет освещенности рабочего места сводится к выбору системы освещения, определению необходимого числа светильников, их типа и размещения. Процесс работы программиста в таких условиях, когда естественное освещение недостаточно или отсутствует. Исходя из этого, рассчитаем параметры искусственного освещения.

Искусственное освещение выполняется посредством электрических источников света двух видов: ламп накаливания и люминесцентных ламп. Будем использовать люминесцентные лампы, которые, по сравнению с лампами накаливания, имеют существенные преимущества:

- по спектральному составу света они близки к дневному, естественному свету.
- обладают более высоким КПД (в 1,5-2 раза выше, чем КПД ламп накаливания).
- обладают повышенной светоотдачей (в 3-4 раза выше, чем у ламп накаливания).
- более длительный срок службы.

Расчет освещения производится для комнаты, длина которой 9 м, ширина - 5 м. Найдем значение освещенности данного помещения и сравним его с нормированным значением, для того, чтобы определить достаточно ли текущего искусственного освещения для работы в помещении.

Вычислим высоту подвеса светильника над рабочей поверхностью

$$H = h - h_p - h_c \quad (5.1)$$

где h_c – расстояние от светильника до перекрытия, $h_c = 0.05$ м;

h_p – высота рабочей поверхности над полом, $h_p = 0.8$ м;

h – высота помещения, $h = 3$ м;

$$H = 3 - 0.8 - 0.05 = 2,15 \text{ м.}$$

Наиболее выгодное расстояние между светильниками определяется как

$$L = \lambda \cdot H, \quad (5.2)$$

где $\lambda = 1.2 \div 1.4$;

$$L = 1.3 \cdot 2.15 = 2,795 \text{ м.}$$

Определяем индекс помещения

$$i = \frac{S}{H \cdot (A + B)}, \quad (5.3)$$

где S - площадь помещения, $S = 45 \text{ м}^2$;

H - расчетная высота подвеса, $H = 2.15 \text{ м}$;

A - ширина помещения, $A = 5 \text{ м}$,

B - длина помещения, $B = 9 \text{ м}$.

Подставив значения получим:

$$i = \frac{45}{2.15 \cdot (5 + 9)} = 1,495$$

Коэффициенты отражения от потолка, стен и пола равны:

$$P_{\text{пот}} = 50\% ;$$

$$P_{\text{ст}} = 10\% ;$$

$$P_{\text{пол}} = 30\% .$$

Зная индекс помещения i находим коэффициент использования $\eta = 40\%$;

K_3 - коэффициент запаса: $K_3 = 1.2$.

Формула для расчета освещенности

$$E = \frac{N \cdot \Phi_{\text{л}} \cdot \eta \cdot n}{S \cdot K_3 \cdot z} \quad (5.4)$$

Для освещения выбираем люминесцентные лампы типа ЛД-65, световой поток которых $\Phi_{\text{л}} = 3750 \text{ Лк}$.

Подставим все значения в формулу (4.4) и получим:

$$E = \frac{7 \cdot 3750 \cdot 2 \cdot 0.4}{45 \cdot 1.2 \cdot 1.1} = 388.89 \text{ Лк}$$

Нормируемая освещенность составляет 400лк, а освещенность в помещении 388.89 Лк. Значит искусственное освещение в помещении в пределах нормы.

Схема расположение светильников

Так как L – наиболее выгодное расстояние между светильниками, l – расстояние от крайних светильников или рядов светильников до стены:

$$l = 0.3 \div 0.5L, \quad (5.5)$$

$$l = 0.3 * 2,795 = 0.8385 \text{ м}$$

Предлагается установить в первой комнате 3 светильника, 2 в ряд, так как длина комнаты 5 метров, а ширина 3метра. А во второй комнате 4 светильника (рисунок 5.2)

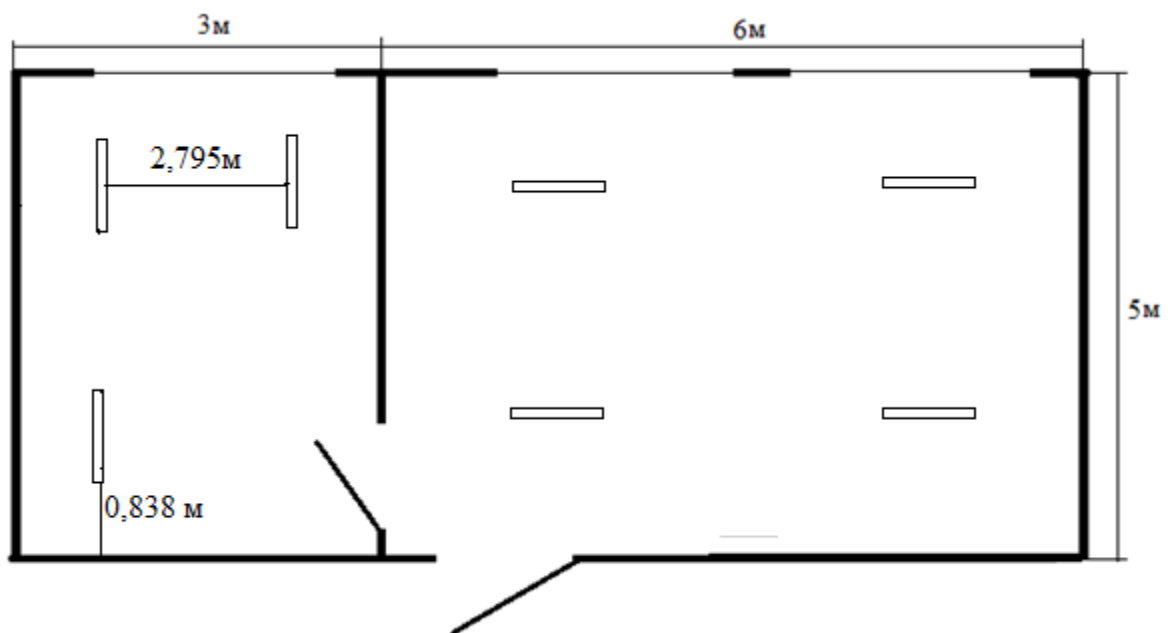


Рисунок 5.1 – Схема расположение светильников

Заключение

В ходе выполнения дипломного проекта была смоделирована АИС «Деятельности преподавателей кафедры «Информационные системы». Разработанная система наглядно демонстрирует основные преимущества визуального моделирования. Моделирование обеспечивает более точную оценку необходимых ресурсов, четкую проработку планов и эффективное функционирование создаваемых систем.

Преимущества:

- данный подход позволяет ускорить процесс разработки ПО;
- наглядность проектируемой модели в процессе разработки, что позволяет легко вносить изменения в модель без редактирования кода.

Данный курсовой проект позволил изучить особенности языка UML, который научил быстро составить представление о технологии, познакомиться с ней на концептуальном и понятийном уровне, а также приобрести навыки работы с пакетом программных продуктов Rational RosE.

На основании подготовленных материалов можно разработать программное обеспечение с рабочим названием автоматизированная информационная система «Деятельности преподавателей кафедры «Информационные системы». После разработки и внедрения данного программного обеспечения можно ожидать следующие результаты:

- снижение объема бумажной документации;
- снижение затрат на хранение бумажной документации;
- автоматизация и ускорение расчета рейтинга всех преподавателей кафедры Информационных систем;
- уменьшение вероятности совершения ошибки при подсчете рейтинга преподавателей;
- сохранность данных в созданной базе данных.

Список использованной литературы

1. Понятие и структура проекта информационной системы. Классификация методов и технологий проектирования ИС. Особенности канонического проектирования информационных систем. Стадии процесса проектирования ИС. [Электронный ресурс] – Режим доступа: <http://csgtr.narod.ru/pis.html>
2. Скоз Е.Ю. Программные средства моделирования в САПР. Конспект лекций.
3. У. Боггс, М. Боггс «UML и Rational RosE 2002» - Издательство «ЛЮРИ», 2004.
4. Шмуллер Д. Освой самостоятельно UML за 24 часа.[Текст]: М.: Издательский дом «Вильямс», 2005г. – 416 с.
5. Вендров А.М. Проектирование программного обеспечения для экономических информационных систем. [Текст]: М:«Финансы и статистика», 2005 г.– 524 с.
6. Введение в UML. Лекции. [Электронный ресурс] – Режим доступа: <http://www.intuit.ru.-30.10.2010>.
7. П.Ноутон, Г.Шилдт«Java 2 Наиболее полное руководство» - Издательство «ExprEs», 2012.
8. Вьюкова Н. Информационная безопасность систем управления базами данных [Электронный ресурс]. – Режим доступа: <http://www.citforum.ru/dataBasE/kBd96/indEx.shtml>.
9. Гладченко А. Создание гибкой системы безопасности [Электронный ресурс]. – Режим доступа: <http://www.sql.ru/articlEs/mssql/01061605.shtml>.
10. Диаграммы UML [Электронный ресурс] – Режим доступа: <http://www.intuit.ru/studiEs/coursEs>
11. Бекишева А. И. Методические указания к выполнению экономической части дипломной работы для бакалавров специальности 5В070300 – Информационные системы – Алматы: АУЭС; 2013. – 24с.
12. Ни А.Г., Имангалиев Ш.И. Методические указания к выполнению дипломных работ (проектов) для студентов специальности 5В070300 «Информационные системы». – Алматы: АУЭС, 2013 – 34 с.
13. Основы UML – Разработка диаграмм в среде RationalRosE [Электронный ресурс] – Режим доступа:<http://2programmEr.ru/uml/>
14. Грабер М. SQL:[пер. с англ.] / М.Грабер.-М: Издательство «Лори», 2003..
15. РД Концепция защиты СВТ и АС от НСД к информации: утв. Гостехкомиссией России 30.03.92. – Москва, 1992
16. Шумейко В. Безопасность информационных сетей и баз данных. Методическое руководство. Составители: В.А. Шумейко, А.О.Башмаков.– Новосибирск: Изд-во НГТУ, 2004. Козленко Л. Информационная безопасность в современных системах управления базами данных /

Л.Козленко // КомпьютерПресс. – 2002. - №3

17. Требования охраны труда при работе на персональных компьютерах. [Электронный ресурс] – Режим доступа: <http://lib.rushkolnik.ru>

ПРИЛОЖЕНИЕ А Техническое задание

А.1 Общие положения

А.1.1 Полное наименование системы и ее условное обозначение

Полное наименование системы: Разработка и проектирование автоматизированной информационной системы «Учет деятельности преподавателей кафедры «Информационные системы». Краткое наименование системы: АИС «Учет деятельности преподавателей кафедры «Информационные системы»

А.1.2 Шифр темы или шифр (номер) договора

Шифр темы: АИС-РРП-ИС-10.

А.1.3 Наименование предприятий (объединений) разработчика и заказчика (пользователя) системы и их реквизиты

Заказчиком системы является кафедра Информационных систем Алматинского университета энергетики и связи

Адрес заказчика: Алматы, ул. А. Байтурсынова, 126, комната Б-331

Телефон: 8 (727) 292-50-95, внутр. 527

Разработчиком системы является студент группы ИС 10-2 Ситниченко Т.В.

А.1.4 Перечень документов, на основании которых создается система, кем и когда утверждены эти документы

Основанием для разработки является Положение АУЭС и ГОСО РК 5.03.015 – 2009

А.1.5 Плановые сроки начала и окончания работы по созданию системы

Плановый срок начала работ по созданию АИС «Деятельности преподавателей кафедры «Информационные системы» – 24 февраля 2014 года.

Плановый срок окончания работ по созданию АИС «Деятельности преподавателей кафедры «Информационные системы» – 27 мая 2014 года

А.1.6 Сведения об источниках и порядке финансирования работ

Создание АИС «Деятельности преподавателей кафедры «Информационные системы» не финансируется.

Продолжение приложения А

А.1.7 Порядок оформления и предъявления заказчику результатов работ

Система передается в виде функционирующего комплекса на базе средств вычислительной техники Заказчика и Исполнителя в сроки, установленные в техническом задании. Приемка системы осуществляется заведующим кафедрой Информационных систем или преподавателями кафедры.

Порядок предъявления системы, ее испытаний и окончательной приемки определен в п.6 настоящего ТЗ. Совместно с предъявлением системы производится сдача разработанного Исполнителем комплекта документации согласно п.8 настоящего ТЗ.

А.1.8 Состав используемой нормативно-технической документации

При разработке автоматизированной управляющей системы и создании проектно-эксплуатационной документации Исполнитель должен руководствоваться требованиями следующих нормативных документов:

ГОСТ 34.601-90. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания;

ГОСТ 34.201-89. Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплексность и обозначение документов при создании автоматизированных систем;

РД 50-34.698-90. Методические указания. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов.

А.2 Назначение и цели создания (развития) системы

– назначение системы – Проектируемая система предназначена для использования её сотрудниками кафедры: заведующим кафедрой, преподавателями и секретарем. Её назначение – расчет рейтинга преподавателей кафедры;

– цели создания системы – снижение трудоемкости обработки, создания, хранения и передачи информации на 70% за счет создания систематизированной базы данных, облегчения подсчета рейтинга и нагрузки за счет автоматизации этих процессов.

А.2.1 Назначение

АИС «Деятельности преподавателей кафедры «Информационные системы» предназначена для автоматизированного расчета рейтинга

Продолжение приложения А

преподавателей кафедры ИС, в частности, для исполнения следующих процессов:

- сбор данных для расчета рейтинга;
- хранение в базе данных информации о преподавателях;
- расчет коэффициентов для подсчета рейтинга;
- вывод информации о рейтинге преподавателя по семестрам и за год;
- хранение и вывод информации о рейтинге преподавателей.

А.2.2 Основными целями создания АИС «Деятельности преподавателей кафедры «Информационные системы» являются:

- сбор информации для расчета;
- автоматизирование расчета рейтинга преподавателя;
- повышение эффективности исполнения расчетов рейтинга путем сокращения непроизводительных и дублирующих операций, операций, выполняемых "вручную", оптимизации информационного взаимодействия участников процессов;
- исключение ошибок при расчетах;
- повышение качества принятия управленческих решений за счет оперативности представления, полноты, достоверности и удобства форматов отображения информации;
- обеспечение целостности, доступности и конфиденциальности информации.

Критерии достижения целей

Для реализации поставленных целей система должна решать следующие задачи:

- создание единой базы данных о деятельности преподавателей кафедры;
- систематизация расчетов рейтинга;
- подготовка отчетов и рейтинга по шаблонам;

Автоматизацию учёта рейтингов и нагрузки, в том числе:

- учёт сроков хранения информации в базе;
- помещение отчетов в базу для последующей работы с ними;
- автоматизацию поиска информации о преподавателе.

А.3 Характеристика объекта автоматизации

А.3.1 Объект автоматизации

Продолжение приложения А

Процессы по управлению АИС «Деятельности преподавателей кафедры «Информационные системы » (п. 2.1), а также контроль эффективности выполнения указанных процессов. Данные процессы осуществляются следующими специалистами:

- Персонал кафедры.

А.3.2 Существующее программное обеспечение

В настоящий момент деятельность кафедры Информационных систем не автоматизирована. Расчет нагрузки и рейтинга преподавателя рассчитывается с помощью программы Excel ответственным за это лицом.

А.3.3 Существующее техническое обеспечение

Телекоммуникационная инфраструктура развернута на базе оборудования, принадлежащего кафедре ИС.

Все серверы БД объединены в единую телекоммуникационную сеть по выделенным линиям с пропускной способностью 1 Мб/сек.

А.3.2 Существующее нормативно-правовое обеспечение

Существующее нормативно-правовое обеспечение составляют федеральные и областные нормативные правовые акты:

- Конституция РК;
- Гражданский кодекс РК.и т.д.

А.4 Требования к системе

Требования к системе в целом:

- требования к структуре и функционированию системы;
- требования к персоналу системы;
- показатели назначения;
- требования к надежности; безопасности; эргономике и технической эстетике; транспортабельности; эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы; защите информации; сохранности информации при авариях; защите от влияния внешних воздействий;

– требования к патентной чистоте; стандартизации и унификации; дополнительные требования.

Требования к функциям (задачам), выполняемым системой; перечень функций, задач или их комплексов, подлежащих автоматизации (по каждой

Продолжение приложения А

подсистеме); очередность ввода в эксплуатацию, временной регламент реализации и требования к качеству реализации каждой функции, задачи (или комплекса задач), к форме представления выходной информации, перечень и критерии отказов для каждой функции, по которой задаются требования по надежности.

Требования к видам обеспечения в зависимости от вида системы приводят требования к математическому, информационному, лингвистическому, программному, техническому, метрологическому, организационному, методическому и другим видам обеспечения системы.

А.4.1 Требования к системе в целом

А.4.1.1 Требования к структуре и функционированию системы

А.4.1.1.1 В составе АИС «Деятельность преподавателей кафедры «Информационные системы» должна решаться задача расчета рейтинга.

Подсистема расчета рейтинга предназначена для расчета рейтинга преподавателя с последующим начислением ему надбавки от администрации университета. Для подсчета рейтинга учитывается ряд коэффициентов, подсчет которых необходим для составления комплексного коэффициента. При расчете рейтинга преподавателя учитываются такие факторы, как результат работы преподавателя за последние 2 года, оценка труда преподавателя другими преподавателями кафедры, его научные достижения, оценка его работы деканом, оценка его работы студентами и оценка его труда в качестве эдвайзера группы.

А.4.1.1.2 Требования к способам и средствам связи для информационного обмена между компонентами системы

А.4.1.1.2 Требования к способам и средствам связи для информационного обмена между компонентами системы

Входящие в состав АИС «Деятельность преподавателей кафедры «Информационные системы» подсистемы в процессе функционирования должны обмениваться информацией на основе открытых форматов обмена данными, используя для этого входящие в их состав модули информационного взаимодействия.

Форматы данных будут разработаны и утверждены на этапе технического проектирования.

А.4.1.1.3 Требования к характеристикам взаимосвязей создаваемой системы со смежными системами

Продолжение приложения А

А.4.1.1.3 Требования к характеристикам взаимосвязей создаваемой системы со смежными системами

А.4.1.1.4 Требования к режимам функционирования системы

Для АИС «Деятельность преподавателей кафедры «Информационные системы» определены следующие режимы функционирования:

Нормальный режим функционирования;

Аварийный режим функционирования.

Основным режимом функционирования АИС является нормальный режим.

В нормальном режиме функционирования системы:

– клиентское программное обеспечение и технические средства пользователей и администратора системы обеспечивают возможность функционирования в течение рабочего дня (с 09:00 до 18:00) шесть дней в неделю;

– серверное программное обеспечение и технические средства серверов обеспечивают возможность круглосуточного функционирования, с перерывами на обслуживание;

– исправно работает оборудование, составляющее комплекс технических средств;

– исправно функционирует системное, базовое и прикладное программное обеспечение системы.

Для обеспечения нормального режима функционирования системы необходимо выполнять требования и выдерживать условия эксплуатации программного обеспечения и комплекса технических средств системы, указанные в соответствующих технических документах (техническая документация, инструкции по эксплуатации и т.д.).

Аварийный режим функционирования системы характеризуется отказом одного или нескольких компонент программного и (или) технического обеспечения.

В случае перехода системы в аварийный режим необходимо:

– завершить работу всех приложений, с сохранением данных;

– выключить рабочие станции операторов;

– выключить все периферийные устройства;

– выполнить резервное копирование БД.

После этого необходимо выполнить комплекс мероприятий по устранению причины перехода системы в аварийный режим.

А.4.1.1.5. Требования по диагностированию системы

Продолжение приложения А

АИС «Деятельности преподавателей кафедры «Информационные системы» должна предоставлять инструменты диагностирования основных процессов системы, трассировки и мониторинга процесса выполнения программы.

Компоненты должны предоставлять удобный интерфейс для возможности просмотра диагностических событий, мониторинга процесса выполнения программ.

При возникновении аварийных ситуаций, либо ошибок в программном обеспечении, диагностические инструменты должны позволять сохранять полный набор информации, необходимой разработчику для идентификации проблемы (снимки экранов, текущее состояние памяти, файловой системы).

А.4.1.1.6 Перспективы развития, модернизации системы

АИС «Деятельность преподавателей кафедры «Информационные системы» должна реализовывать возможность дальнейшей модернизации как программного обеспечения.

А.4.1.2 Требования к численности и квалификации персонала системы

Для эксплуатации АИС «Деятельность преподавателей кафедры «Информационные системы» определены следующие роли:

- секретарь (системный администратор);
- администратор баз данных;
- администратор информационной безопасности;
- пользователи (преподаватели и заведующий кафедры).

Основными обязанностями секретаря, который по сути будет выполнять работу системного администратора, являются:

- модернизация, настройка и мониторинг работоспособности комплекса технических средств (серверов, рабочих станций);
- установка, модернизация, настройка и мониторинг работоспособности системного и базового программного обеспечения;
- установка, настройка и мониторинг прикладного программного обеспечения;
- ведение учетных записей пользователей системы.

Секретарь должен обладать высоким уровнем квалификации и практическим опытом выполнения работ по установке, настройке и администрированию программных и технических средств, применяемых в системе.

Основными обязанностями администратора баз данных являются:

- установка, модернизация, настройка параметров программного обеспечения СУБД;

Продолжение приложения А

- оптимизация прикладных баз данных по времени отклика, скорости доступа к данным;
- разработка, управление и реализация эффективной политики доступа к информации, хранящейся в прикладных базах данных.
- администратор баз данных должен обладать высоким уровнем квалификации и практическим опытом выполнения работ по установке, настройке и администрированию используемых в АС СУБД.
- основными обязанностями администратора информационной безопасности являются:
 - разработка, управление и реализация эффективной политики информационной безопасности системы;
 - управление правами доступа пользователей к функциям системы;
 - осуществление мониторинга информационной безопасности.

Администратор информационной безопасности данных должен обладать высоким уровнем квалификации и практическим опытом выполнения работ по обеспечению информационной безопасности.

Пользователи системы должны иметь опыт работы с персональным компьютером на базе операционных систем Microsoft Windows на уровне квалифицированного пользователя и свободно осуществлять базовые операции в стандартных Windows.

Роли системного администратора, администратора баз данных и администратора информационной безопасности могут быть совмещены в роль секретаря либо лаборанта кафедры.

Рекомендуемая численность для эксплуатации АИС «Деятельности преподавателей кафедры «Информационные системы»:

Администратор - 1 штатная единица;

Пользователь - число штатных единиц определяется структурой кафедры.

А.4.1.3 Показатели назначения

АИС «Деятельность преподавателей кафедры «Информационные системы» должны обеспечивать возможность исторического хранения данных не менее 10 лет.

Система должна обеспечивать возможность одновременной работы не менее 10 пользователей для подсистемы операционной деятельности, и не менее 5-ти пользователей для других подсистем при следующих характеристиках времени отклика системы:

для операций навигации по экранным формам системы - не более 5 сек;

для операций формирования справок и выписок - не более 10 сек.

Время формирования аналитических отчетов определяется их сложностью и может занимать продолжительное время.

Продолжение приложения А

Система должна предусматривать возможность масштабирования по производительности и объему обрабатываемой информации без модификации ее программного обеспечения путем модернизации используемого комплекса технических средств. Возможности масштабирования должны обеспечиваться средствами используемого базового программного обеспечения.

А.4.1.4 Требования к надежности

Система должна сохранять работоспособность и обеспечивать восстановление своих функций при возникновении следующих внештатных ситуаций:

- при сбоях в системе электроснабжения аппаратной части, приводящих к перезагрузке ОС, восстановление программы должно происходить после перезапуска ОС и запуска исполняемого файла системы;
- при ошибках в работе аппаратных средств (кроме носителей данных и программ) восстановление функции системы возлагается на ОС;
- при ошибках, связанных с программным обеспечением (ОС и драйверы устройств), восстановление работоспособности возлагается на ОС.

Для защиты аппаратуры от бросков напряжения и коммутационных помех должны применяться сетевые фильтры.

А.4.1.5 Требования к безопасности

Все внешние элементы технических средств системы, находящиеся под напряжением, должны иметь защиту от случайного прикосновения, а сами технические средства иметь зануление или защитное заземление в соответствии с ГОСТ 12.1.030-81 и ПУЭ.

Система электропитания должна обеспечивать защитное отключение при перегрузках и коротких замыканиях в цепях нагрузки, а также аварийное ручное отключение.

Общие требования пожарной безопасности должны соответствовать нормам на бытовое электрооборудование. В случае возгорания не должно выделяться ядовитых газов и дымов. После снятия электропитания должно быть допустимо применение любых средств пожаротушения.

Факторы, оказывающие вредные воздействия на здоровье со стороны всех элементов системы (в том числе инфракрасное, ультрафиолетовое, рентгеновское и электромагнитное излучения, вибрация, шум, электростатические поля, ультразвук строчной частоты и т.д.), не должны превышать действующих норм (СанПиН 2.2.2./2.4.1340-03 от 03.06.2003 г.).

А.4.1.6 Требования к эргономике и технической эстетике

Продолжение приложения А

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав системы должно осуществляться посредством визуального графического интерфейса (GUI). Интерфейс системы должен быть понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм. Навигационные элементы должны быть выполнены в удобной для пользователя форме. Средства редактирования информации должны удовлетворять принятым соглашениям в части использования функциональных клавиш, режимов работы, поиска, использования оконной системы. Ввод-вывод данных системы, прием управляющих команд и отображение результатов их исполнения должны выполняться в интерактивном режиме. Интерфейс должен соответствовать современным эргономическим требованиям и обеспечивать удобный доступ к основным функциям и операциям системы.

Интерфейс должен быть рассчитан на преимущественное использование манипулятора типа "мышь", то есть управление системой должно осуществляться с помощью набора экранных меню, кнопок, значков и т. п. элементов. Клавиатурный режим ввода должен использоваться главным образом при заполнении и/или редактировании текстовых и числовых полей экранных форм.

Все надписи экранных форм, а также сообщения, выдаваемые пользователю (кроме системных сообщений) должны быть на русском языке.

Система должна обеспечивать корректную обработку аварийных ситуаций, вызванных неверными действиями пользователей, неверным форматом или недопустимыми значениями входных данных. В указанных случаях система должна выдавать пользователю соответствующие сообщения, после чего возвращаться в рабочее состояние, предшествовавшее неверной (недопустимой) команде или некорректному вводу данных.

Система должна соответствовать требованиям эргономики и профессиональной медицины при условии комплектования высококачественным оборудованием (ПЭВМ, монитор и прочее оборудование), имеющим необходимые сертификаты соответствия и безопасности

А.4.1.7 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

Система должна быть рассчитана на эксплуатацию в составе программно-технического комплекса Заказчика и учитывать разделение ИТ инфраструктуры Заказчика на внутреннюю и внешнюю. Техническая и физическая защита аппаратных компонентов системы, носителей данных, бесперебойное энергоснабжение, резервирование ресурсов, текущее

Продолжение приложения А

обслуживание реализуется техническими и организационными средствами, предусмотренными в ИТ инфраструктуре Заказчика.

Для нормальной эксплуатации разрабатываемой системы должно быть обеспечено бесперебойное питание ПЭВМ. При эксплуатации система должна быть обеспечена соответствующая стандартам хранения носителей и эксплуатации ПЭВМ температура и влажность воздуха.

Периодическое техническое обслуживание используемых технических средств должно проводиться в соответствии с требованиями технической документации изготовителей, но не реже одного раза в год.

Периодическое техническое обслуживание и тестирование технических средств должны включать в себя обслуживание и тестирование всех используемых средств, включая рабочие станции, серверы, кабельные системы и сетевое оборудование, устройства бесперебойного питания.

В процессе проведения периодического технического обслуживания должны проводиться внешний и внутренний осмотр и чистка технических средств, проверка контактных соединений, проверка параметров настроек работоспособности технических средств и тестирование их взаимодействия.

Восстановление работоспособности технических средств должно проводиться в соответствии с инструкциями разработчика и поставщика технических средств и документами по восстановлению работоспособности технических средств и завершаться проведением их тестирования. Размещение помещений и их оборудование должны исключать возможность бесконтрольного проникновения в них посторонних лиц и обеспечивать сохранность находящихся в этих помещениях конфиденциальных документов и технических средств.

Размещение оборудования, технических средств должно соответствовать требованиям техники безопасности, санитарным нормам и требованиям пожарной безопасности.

Все пользователи системы должны соблюдать правила эксплуатации электронной вычислительной техники.

Квалификация персонала и его подготовка должны соответствовать технической документации.

А.4.1.8 Требования к защите информации от несанкционированного доступа

АИС должна обеспечивать защиту от несанкционированного доступа (НСД).

Компоненты подсистемы защиты от НСД должны обеспечивать:

- идентификацию пользователя;
- проверку полномочий пользователя при работе с системой;

Продолжение приложения А

– разграничение доступа пользователей на уровне задач и информационных массивов.

Протоколы аудита системы и приложений должны быть защищены от несанкционированного доступа как локально, так и в архиве.

Защищённая часть системы должна использовать "слепые" пароли (при наборе пароля его символы не показываются на экране либо заменяются одним типом символов; количество символов не соответствует длине пароля).

Защищённая часть системы должна автоматически блокировать сессии пользователей и приложений по заранее заданным временам отсутствия активности со стороны пользователей и приложений.

Защищённая часть системы должна использовать многоуровневую систему защиты. Защищённая часть системы должна быть отделена от незащищённой части системы межсетевым экраном.

А.4.1.9 Требования по сохранности информации при авариях

Программное обеспечение АИС «Деятельность преподавателей кафедры «Информационные системы» должно восстанавливать свое функционирование при корректном перезапуске аппаратных средств. Должна быть предусмотрена возможность организации автоматического и (или) ручного резервного копирования данных системы средствами системного и базового программного обеспечения (ОС, СУБД), входящего в состав программно-технического комплекса Заказчика.

Приведенные выше требования не распространяются на компоненты системы, разработанные третьими сторонами и действительны только при соблюдении правил эксплуатации этих компонентов, включая своевременную установку обновлений, рекомендованных производителями покупного программного обеспечения.

А.4.1.10 Требования к защите от влияния внешних воздействий

А.4.1.11 Требования к патентной чистоте

Установка системы в целом, как и установка отдельных частей системы не должна предъявлять дополнительных требований к покупке лицензий на программное обеспечение сторонних производителей, кроме программного обеспечения, указанного в разделе.

А.4.1.12 Требования по стандартизации и унификации

Экранные формы должны проектироваться с учетом требований унификации:

Продолжение приложения А

- все экранные формы пользовательского интерфейса должны быть выполнены в едином графическом дизайне, с одинаковым расположением основных элементов управления и навигации;
- для обозначения сходных операций должны использоваться сходные графические значки, кнопки и другие управляющие (навигационные) элементы. Термины, используемые для обозначения типовых операций (добавление информационной сущности, редактирование поля данных), а также последовательности действий пользователя при их выполнении, должны быть унифицированы;
- внешнее поведение сходных элементов интерфейса (реакция на наведение указателя "мыши", переключение фокуса, нажатие кнопки) должны реализовываться одинаково для однотипных элементов.

А.4.2 Требования к функциям (задачам), выполняемым системой

А.4.2.1 Подсистема расчета рейтинга преподавателя

Данная подсистема должна осуществлять перенос информации, которая будет требоваться персоналом деканата для составления отчетов и форм, таких как 1-к, 3-нк и формы 34.

А.4.3 Требования к видам обеспечения

А.4.3.1 Требования к математическому обеспечению системы

Математические методы и алгоритмы, используемые для шифрования/дешифрования данных, а также программное обеспечение, реализующее их, должны быть сертифицированы уполномоченными организациями

А.4.3.2 Требования к информационному обеспечению системы

Состав, структура и способы организации данных в системе должны быть определены на этапе технического проектирования.

Хранение данных должно осуществляться на основе современных реляционных или СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД.

Средства СУБД, а также средства используемых операционных систем должны обеспечивать документирование и протоколирование обрабатываемой в системе информации.

Структура базы данных должна поддерживать кодирование хранимой и обрабатываемой информации в соответствии с общероссийскими классификаторами (там, где они применимы).

Продолжение приложения А

Доступ к данным должен быть предоставлен только авторизованным пользователям с учетом их служебных полномочий, а также с учетом категории запрашиваемой информации.

Структура базы данных должна быть организована рациональным способом, исключающим одновременную полную выгрузку информации, содержащейся в базе данных системы.

Технические средства, обеспечивающие хранение информации, должны использовать современные технологии, позволяющие обеспечить повышенную надежность хранения данных и оперативную замену оборудования (распределенная избыточная запись/считывание данных; зеркалирование; независимые дисковые массивы; кластеризация).

В состав системы должна входить специализированная подсистема резервного копирования и восстановления данных.

При проектировании и развертывании системы необходимо рассмотреть возможность использования накопленной информации из уже функционирующих информационных систем. Перечень функционирующих информационных систем приведен в разделе 3 настоящего документа.

А.4.3.3 Требования к лингвистическому обеспечению системы

Все прикладное программное обеспечение системы для организации взаимодействия с пользователем должно использовать русский язык.

А.4.3.4 Требования к программному обеспечению системы

При проектировании и разработке системы необходимо максимально эффективным образом использовать ранее закупленное программное обеспечение, как серверное, так и для рабочих станций.

Используемое при разработке программное обеспечение и библиотеки программных кодов должны иметь широкое распространение, быть общедоступными и использоваться в промышленных масштабах. Базовой программной платформой должна являться операционная система MS Windows.

А.4.3.5 Требования к техническому обеспечению

Техническое обеспечение системы должно максимально и наиболее эффективным образом использовать существующие в органах федерального агентства технические средства.

В состав комплекса должны следующие технические средства:

- серверы БД;
- серверы приложений;

Продолжение приложения А

- сервер системы формирования отчетности;
- веб сервер;
- ПК пользователей;
- ПК администраторов.

А.4.3.6 Требования к метрологическому обеспечению

А.4.3.7 Требования к организационному обеспечению

Организационное обеспечение системы должно быть достаточным для эффективного выполнения персоналом возложенных на него обязанностей при осуществлении автоматизированных и связанных с ними неавтоматизированных функций системы.

Заказчиком должны быть определены должностные лица, ответственные за:

- обработку информации АИС;
- администрирование АИС;
- обеспечение безопасности информации АИС;
- управление работой персонала по обслуживанию АИС.

К работе с системой должны допускаться сотрудники, имеющие навыки работы на персональном компьютере, ознакомленные с правилами эксплуатации и прошедшие обучение работе с системой.

А.4.3.8 Требования к методическому обеспечению

В состав методического обеспечения системы должны входить следующие законодательные акты, стандарты, нормативы, инструкции.

А.5 Состав и содержание работ по созданию (развитию) системы

Раздел "Состав и содержание работ по созданию (развитию) системы" должен содержать перечень стадий и этапов работ по созданию системы в соответствии с ГОСТ, сроки их выполнения, перечень организаций - исполнителей работ, ссылки на документы, подтверждающие согласие этих организаций на участие в создании системы, или запись, определяющую ответственного (заказчик или разработчик) за проведение этих работ.

А.6 Порядок контроля и приемки системы

А.6.1 Виды, состав, объем и методы испытаний системы

Виды, состав, объем, и методы испытаний подсистемы должны быть изложены в программе и методике испытаний АИС «Деятельность

Продолжение приложения А

преподавателей кафедры «Информационные системы», разрабатываемой в составе рабочей документации.

А.6.2 Общие требования к приемке работ по стадиям

Сдача-приёмка работ производится поэтапно, в соответствии с рабочей программой и календарным планом.

Сдача-приёмка осуществляется комиссией, в состав которой входят представители Заказчика и Исполнителя.

Все создаваемые в рамках настоящей работы программные изделия (за исключением покупных) передаются Заказчику, как в виде готовых модулей, так и в виде исходных кодов, представляемых в электронной форме на стандартном машинном носителе (например, на компакт-диске).

А.6.3 Статус приемочной комиссии

Статус приемочной комиссии определяется Заказчиком до проведения испытаний.

А.7 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

В ходе выполнения проекта на объекте автоматизации требуется выполнить работы по подготовке к вводу системы в действие. При подготовке к вводу в эксплуатацию АИС «Деятельность преподавателей кафедры «Информационные системы» Заказчик должен обеспечить выполнение следующих работ:

Определить ответственных должностных лиц, ответственных за внедрение и проведение опытной эксплуатации АИС «Деятельность преподавателей кафедры «Информационные системы»;

Обеспечить присутствие пользователей на обучении работе с системой, проводимом Исполнителем;

Совместно с Исполнителем подготовить план развертывания системы на технических средствах Заказчика;

Провести опытную эксплуатацию АИС «Деятельность преподавателей кафедры «Информационные системы».

А.8 Требования к документированию

Данный проект сопровождается ТЗ и документацией на технический проект.

В техническом задании расписываются:

Продолжение приложения А

- основные цели, задачи, сроки и этапы разработки;
- список основных функций и требований;
- список функций интерфейса.

Документация на технический проект является инструкцией по использованию данного ПО. В данной документации будут расписываться:

- условия работы ПО;
- установка ПО;
- использование ПО, попунктное описание основных функций и функций интерфейса;
- устранение проблем при переходе в аварийный режим.

Для системы на различных стадиях создания должны быть выпущены следующие документы из числа предусмотренных в ГОСТ 34.201-«Информационная технология. Комплекс стандартов на автоматизированные системы».

А.9 Источники разработки

Документы и информационные материалы (технико-экономическое обоснование, отчеты о законченных научно-исследовательских работах, информационные материалы на отечественные, зарубежные системы-аналоги и др.), на основании которых разрабатывалось ТЗ и которые должны быть использованы при создании системы.

Технико-экономическое обоснование. Данный документ содержит, финансовое описание системы, в котором содержится перечень используемых ресурсов и их ценовое описание. Стоимость системы, подсчет рентабельности. Минимизация используемых ресурсов для получения наиболее максимальной прибыли.

Данная система должна разрабатываться на основании ТК 34 по стандартизации Информационные технологии. Номер приказа и дата утверждения: от 27.07.01 г. № 274.

ПРИЛОЖЕНИЕ Б Код программы

Первая форма

```
#include <vcl.h>
#pragma hdrstop

#include "Unit1.h"
#include "Unit2.h"
#include "Unit3.h"
#include "Unit4.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponEnt* OwnEr)
    : TForm(OwnEr)
{
}
//-----
void __fastcall TForm1::Button1Click(TObjEct *SEndEr)
{
    Form1->Close();
}
//-----
void __fastcall TForm1::Button2Click(TObjEct *SEndEr)
{
    ADOQuEry1->Active = false;
    ADOQuEry1->Active = true;
}
//-----

void __fastcall TForm1::DBGrid1DBClick(TObjEct *SEndEr)
{
    Form4->Show();
    Form1->Visible = false;
}
//-----

void __fastcall TForm1::Button3Click(TObjEct *SEndEr)
{
    Form4->Show();
}
```

Продолжение приложения Б

```
}
```

Вторая форма

```
#include <vcl.h>
#pragma hdrstop

#include "Unit10.h"
#include "Unit8.h"
#include "Unit6.h"
//-----
#pragma packagE(smart_init)
#pragma rEsourcE "*.dfm"
TForm10 *Form10;
//-----
__fastcall TForm10::TForm10(TComponEnt* OwnEr)
    : TForm(OwnEr)
{
}
//-----
void __fastcall TForm10::Button1Click(TObjEct *SEndEr)
{
    Form6->Show();
    Form10->ClosE();
}
```

Третья форма

```
#include <vcl.h>
#pragma hdrstop

#include "Unit2.h"
#include "Unit3.h"
//-----
#pragma packagE(smart_init)
#pragma rEsourcE "*.dfm"
TForm2 *Form2;
//-----
__fastcall TForm2::TForm2(TComponEnt* OwnEr)
    : TForm(OwnEr)
{
}
```

Продолжение приложения Б

```
//-----  
void __fastcall TForm2::Button1Click(TObject *SEndEr)  
{  
    Form3->Visible = true;  
    Form2->Close();  
}
```

Четвертая форма

```
#include <vcl.h>  
#pragma hdrstop  
  
#include "Unit3.h"  
#include "Unit1.h"  
#include "Unit2.h"  
//-----  
#pragma package(smart_init)  
#pragma resource "*.dfm"  
TForm3 *Form3;  
//-----  
__fastcall TForm3::TForm3(TComponent* Owner)  
    : TForm(Owner)  
{  
}  
//-----  
void __fastcall TForm3::Button1Click(TObject *SEndEr)  
{  
    int a,b,c,d,e,f,g,h,r;  
    if ((Edit1->Text == "") || (Edit2->Text == "") || (Edit3->Text == "") ||  
(Edit4->Text == "") || (Edit5->Text == "") || (Edit6->Text == "") || (Edit7->Text  
== "") || (Edit8->Text == "") || (Edit9->Text == ""))  
    {  
        ShowMessage("Заполните данные");  
    }  
    else {  
        Edit10->Text = IntToStr(StrToInt(DBEdit1->Text) * StrToInt(Edit1-  
>Text));  
        Edit11->Text = IntToStr(StrToInt(DBEdit2->Text) * StrToInt(Edit2-  
>Text));  
        Edit12->Text = IntToStr(StrToInt(DBEdit3->Text) * StrToInt(Edit3-  
>Text));  
        Edit13->Text = IntToStr(StrToInt(DBEdit4->Text) * StrToInt(Edit4-  
>Text));
```

Продолжение приложения Б

```
    Edit14->TExt = IntToStr(StrToInt(DBEdit5->TExt) * StrToInt(Edit5-
>TExt));
    Edit15->TExt = IntToStr(StrToInt(DBEdit6->TExt) * StrToInt(Edit6-
>TExt));
    Edit16->TExt = IntToStr(StrToInt(DBEdit7->TExt) * StrToInt(Edit7-
>TExt));
    Edit17->TExt = IntToStr(StrToInt(DBEdit8->TExt) * StrToInt(Edit8-
>TExt));
    Edit18->TExt = IntToStr(StrToInt(DBEdit9->TExt) * StrToInt(Edit9-
>TExt));
    a = StrToInt(Edit10->TExt);
    B = StrToInt(Edit11->TExt);
    c = StrToInt(Edit12->TExt);
    d = StrToInt(Edit13->TExt);
    E = StrToInt(Edit14->TExt);
    f = StrToInt(Edit15->TExt);
    g = StrToInt(Edit16->TExt);
    h = StrToInt(Edit17->TExt);
    r = StrToInt(Edit18->TExt);
    Edit19->TExt = 1 + IntToStr((a+B+c+d+E+f+g-h-r))/20;
    DBEdit10->TExt == " ";
    DBEdit10->Visible = true;
    Label57->Visible = true;
    ShowMessage("Заполните окно - Введите итоговый балл");
}
}
//-----
void __fastcall TForm3::Button2Click(TObject *SEnder)
{
    Edit1->Clear();
    Edit2->Clear();
    Edit3->Clear();
    Edit4->Clear();
    Edit5->Clear();
    Edit6->Clear();
    Edit7->Clear();
    Edit8->Clear();
    Edit9->Clear();
    Edit10->Clear();
    Edit11->Clear();
    Edit12->Clear();
    Edit13->Clear();
}
```


Продолжение приложения Б

Edit14->Clear();
Edit15->Clear();
Edit16->Clear();
Edit17->Clear();
Edit18->Clear();
Edit19->Clear();
Edit20->Clear();
Edit21->Clear();
Edit22->Clear();
Edit23->Clear();
Edit23->Clear();
Edit24->Clear();
Edit25->Clear();
Edit26->Clear();
Edit27->Clear();
Edit28->Clear();
Edit29->Clear();
Edit30->Clear();
Edit31->Clear();
Edit32->Clear();
Edit33->Clear();
Edit34->Clear();
Edit35->Clear();
Edit36->Clear();
Edit37->Clear();
Edit38->Clear();
Edit39->Clear();
Edit40->Clear();
Edit41->Clear();
Edit42->Clear();
Edit43->Clear();
Edit44->Clear();
Edit45->Clear();
Edit46->Clear();
Edit47->Clear();
Edit48->Clear();
Edit49->Clear();
Edit50->Clear();
Edit51->Clear();
Edit52->Clear();
Edit53->Clear();
Edit54->Clear();

Продолжение приложения Б

```
Edit55->ClEar();  
Edit56->ClEar();  
Edit57->ClEar();  
Edit58->ClEar();  
Edit59->ClEar();  
Edit60->ClEar();  
Edit61->ClEar();  
Edit62->ClEar();  
Edit63->ClEar();  
Edit64->ClEar();  
Edit65->ClEar();  
Edit66->ClEar();  
Edit67->ClEar();  
Edit68->ClEar();  
Edit69->ClEar();  
Edit70->ClEar();  
Edit71->ClEar();  
Edit72->ClEar();  
Edit73->ClEar();  
Edit74->ClEar();  
Edit75->ClEar();  
Edit76->ClEar();  
Edit77->ClEar();  
Edit78->ClEar();  
Edit79->ClEar();  
Edit80->ClEar();  
Edit81->ClEar();  
Edit82->ClEar();  
Edit83->ClEar();  
Edit84->ClEar();  
Edit85->ClEar();  
Edit86->ClEar();  
Edit87->ClEar();  
Edit88->ClEar();  
Edit89->ClEar();  
Edit90->ClEar();  
Edit91->ClEar();  
Edit92->ClEar();  
Edit93->ClEar();  
Edit94->ClEar();  
Form1->Show();  
Form3->ClosE();
```

Продолжение приложения Б

```
}
//-----
void __fastcall TForm3::Button3Click(TObject *SEndEr)
{
    Form2->Show();
    ADOQuery1->Active = false;
    ADOQuery1->SQL->Clear();
    ADOQuery1->SQL->Add("Select п.Фамилия, п.Имя, п.Отчество,
б.К1_Р МР, б.К2_Р ДСК, б.К3_Р НД, б.К4_Р ОПД, б.К5_Р ЗАМ, б.К6_Р
ОПС, б.К_Рейтинг");
    ADOQuery1->SQL->Add("FROM Баллы б, Рейтинг р,
Преподаватели п");
    ADOQuery1->SQL->Add("where п.Фамилия Like('%"+Form1-
>DBEdit7->Text+"%') and п.id_Препод = р.id_Препод and б.id_Бал = р.id_Бал
");
    ADOQuery1->Active = true;
    Form3->Visible = false;
}
//-----

void __fastcall TForm3::Button4Click(TObject *SEndEr)
{
    int
q,w,E,r,t,y,u,i,o,p,a,s,d,f,g,h,j,k,l,z,x,c,v,B,n,m,qw,qE,qr,qt,qy,qu,qi,qo,qp,qa,qS,qd
;
    Edit57->Clear();
    // Проверка на пустоту Едитов
    if ((Edit20->Text == "") || (Edit21->Text == "") || (Edit22->Text == "") ||
(Edit23->Text == "") || (Edit24->Text == "") || (Edit25->Text == "") || (Edit26-
>Text == "") || (Edit27->Text == "") || (Edit28->Text == "") || (Edit29->Text ==
"") || (Edit30->Text == "") || (Edit31->Text == "") || (Edit43->Text == "") ||
(Edit32->Text == "") || (Edit33->Text == "") || (Edit34->Text == "") || (Edit35-
>Text == "") || (Edit36->Text == "") || (Edit37->Text == "") || (Edit38->Text ==
"") || (Edit39->Text == "") || (Edit40->Text == "") || (Edit41->Text == "") ||
(Edit42->Text == "") || (Edit55->Text == "") || (Edit44->Text == "") || (Edit45-
>Text == "") || (Edit46->Text == "") || (Edit47->Text == "") || (Edit56->Text ==
"") || (Edit48->Text == "") || (Edit49->Text == "") || (Edit50->Text == "") ||
(Edit51->Text == "") || (Edit52->Text == "") || (Edit53->Text == "") || (Edit54-
>Text == ""))
    {
        ShowMessage("Заполните данные");
    }
}
```

Продолжение приложения Б

```
}ElsE {  
  //Первый столбец  
  Edit58->TExt = IntToStr(StrToInt(DBEdit11->TExt) * StrToInt(Edit20->TExt));  
  Edit59->TExt = IntToStr(StrToInt(DBEdit12->TExt) * StrToInt(Edit21->TExt));  
  Edit60->TExt = IntToStr(StrToInt(DBEdit13->TExt) * StrToInt(Edit22->TExt));  
  Edit61->TExt = IntToStr(StrToInt(DBEdit35->TExt) * StrToInt(Edit23->TExt));  
  Edit62->TExt = IntToStr(StrToInt(DBEdit36->TExt) * StrToInt(Edit24->TExt));  
  Edit63->TExt = IntToStr(StrToInt(DBEdit37->TExt) * StrToInt(Edit25->TExt));  
  Edit64->TExt = IntToStr(StrToInt(DBEdit38->TExt) * StrToInt(Edit26->TExt));  
  Edit65->TExt = IntToStr(StrToInt(DBEdit39->TExt) * StrToInt(Edit27->TExt));  
  Edit66->TExt = IntToStr(StrToInt(DBEdit40->TExt) * StrToInt(Edit28->TExt));  
  Edit67->TExt = IntToStr(StrToInt(DBEdit41->TExt) * StrToInt(Edit29->TExt));  
  Edit68->TExt = IntToStr(StrToInt(DBEdit42->TExt) * StrToInt(Edit30->TExt));  
  Edit69->TExt = IntToStr(StrToInt(DBEdit43->TExt) * StrToInt(Edit31->TExt));  
  //Второй столбец  
  Edit70->TExt = IntToStr(StrToInt(DBEdit44->TExt) * StrToInt(Edit43->TExt));  
  Edit71->TExt = IntToStr(StrToInt(DBEdit45->TExt) * StrToInt(Edit32->TExt));  
  Edit72->TExt = IntToStr(StrToInt(DBEdit46->TExt) * StrToInt(Edit33->TExt));  
  Edit73->TExt = IntToStr(StrToInt(DBEdit14->TExt) * StrToInt(Edit34->TExt));  
  Edit74->TExt = IntToStr(StrToInt(DBEdit15->TExt) * StrToInt(Edit35->TExt));  
  Edit75->TExt = IntToStr(StrToInt(DBEdit16->TExt) * StrToInt(Edit36->TExt));  
  Edit76->TExt = IntToStr(StrToInt(DBEdit17->TExt) * StrToInt(Edit37->TExt));
```

Продолжение приложения Б

```
    Edit77->TExt = IntToStr(StrToInt(DBEdit18->TExt) * StrToInt(Edit38-
>TExt));

    Edit78->TExt = IntToStr(StrToInt(DBEdit19->TExt) * StrToInt(Edit39-
>TExt));

    Edit79->TExt = IntToStr(StrToInt(DBEdit20->TExt) * StrToInt(Edit40-
>TExt));
    Edit80->TExt = IntToStr(StrToInt(DBEdit21->TExt) * StrToInt(Edit41-
>TExt));
    Edit81->TExt = IntToStr(StrToInt(DBEdit22->TExt) * StrToInt(Edit42-
>TExt));
    //Трейтий столбец
    Edit82->TExt = IntToStr(StrToInt(DBEdit23->TExt) * StrToInt(Edit55-
>TExt));
    Edit83->TExt = IntToStr(StrToInt(DBEdit24->TExt) * StrToInt(Edit44-
>TExt));
    Edit84->TExt = IntToStr(StrToInt(DBEdit25->TExt) * StrToInt(Edit45-
>TExt));
    Edit85->TExt = IntToStr(StrToInt(DBEdit26->TExt) * StrToInt(Edit46-
>TExt));
    Edit86->TExt = IntToStr(StrToInt(DBEdit27->TExt) * StrToInt(Edit47-
>TExt));
    Edit87->TExt = IntToStr(StrToInt(DBEdit47->TExt) * StrToInt(Edit56-
>TExt));
    Edit88->TExt = IntToStr(StrToInt(DBEdit28->TExt) * StrToInt(Edit48-
>TExt));
    Edit89->TExt = IntToStr(StrToInt(DBEdit29->TExt) * StrToInt(Edit49-
>TExt));
    Edit90->TExt = IntToStr(StrToInt(DBEdit30->TExt) * StrToInt(Edit50-
>TExt));
    Edit91->TExt = IntToStr(StrToInt(DBEdit31->TExt) * StrToInt(Edit51-
>TExt));
    Edit92->TExt = IntToStr(StrToInt(DBEdit32->TExt) * StrToInt(Edit52-
>TExt));
    Edit93->TExt = IntToStr(StrToInt(DBEdit33->TExt) * StrToInt(Edit53-
>TExt));
    Edit94->TExt = IntToStr(StrToInt(DBEdit34->TExt) * StrToInt(Edit54-
>TExt));
    // Присвоение значений ЕДИТОВ буквам
```

Продолжение листинга Б

```
q = StrToInt(Edit58->TExt) + StrToInt(Edit59->TExt) + StrToInt(Edit60->TExt) + StrToInt(Edit61->TExt) + StrToInt(Edit62->TExt) + StrToInt(Edit63->TExt) + StrToInt(Edit64->TExt) + StrToInt(Edit65->TExt);
o = StrToInt(Edit66->TExt) + StrToInt(Edit67->TExt) + StrToInt(Edit68->TExt) + StrToInt(Edit69->TExt) + StrToInt(Edit70->TExt) + StrToInt(Edit71->TExt) + StrToInt(Edit72->TExt) + StrToInt(Edit73->TExt);
j = StrToInt(Edit74->TExt) + StrToInt(Edit75->TExt) + StrToInt(Edit76->TExt) + StrToInt(Edit77->TExt) + StrToInt(Edit78->TExt) + StrToInt(Edit79->TExt) + StrToInt(Edit80->TExt) + StrToInt(Edit81->TExt);
m = StrToInt(Edit82->TExt) + StrToInt(Edit83->TExt) + StrToInt(Edit84->TExt) + StrToInt(Edit85->TExt) + StrToInt(Edit86->TExt) + StrToInt(Edit87->TExt) + StrToInt(Edit88->TExt) + StrToInt(Edit89->TExt);
qo= StrToInt(Edit90->TExt) + StrToInt(Edit91->TExt) + StrToInt(Edit92->TExt) + StrToInt(Edit93->TExt) + StrToInt(Edit94->TExt);
// Итоговый подсчет
Edit57->TExt = IntToStr(q + o + j + m + qo)/20;
DBEdit48->TExt == " ";
DBEdit48->Visible = true;
Label58->Visible = true;
ShowMessage("Заполните окно - Введите итоговый балл");
}
}
//-----
void __fastcall TForm3::DBNavigator1BeforeAction(TObject *Sender,
TNavigatorBtn Button)
{
    if (DBEdit10->TExt != Edit19->TExt) {
        ShowMessage("Не верное значение или символ!");
        DBNavigator1->Enabled = false;
        Button2->Visible = false;
        Button3->Visible = false;
    }
    Else {
        ShowMessage("Данные добавлены в базу!");
        DBNavigator1->Enabled = true;
        Button2->Visible = true;
        Button3->Visible = true;
    }
}
//-----
```

Продолжение листинга Б

```
void __fastcall TForm3::DBNavigator2BEforEAction(TObject *SEndEr,
TNavigatorBtn Button)
{
    if (DBEdit48->TEdit != Edit57->TEdit) {
        ShowMessage("Не верное значение или символ!");
        DBNavigator2->Enabled = false;
        Button2->Visible = false;
        Button3->Visible = false;
    }
    Else {
        ShowMessage("Данные добавлены в базу!");
        DBNavigator2->Enabled = true;
        Button2->Visible = true;
        Button3->Visible = true;
    }
}
//-----
```

```
void __fastcall TForm3::DBEdit10Change(TObject *SEndEr)
{
    DBNavigator1->Enabled = true;
}
//-----
```

```
void __fastcall TForm3::DBEdit48Change(TObject *SEndEr)
{
    DBNavigator2->Enabled = true;
}
```

Пятая форма

```
#include <vcl.h>
#pragma hdrstop
```

```
#include "Unit4.h"
#include "Unit1.h"
#include "Unit3.h"
#include "Unit5.h"
#include "Unit6.h"
#include "Unit7.h"
```

```
//-----
#pragma package(smart_init)
```

Продолжение приложения Б

```
#pragma rEsourcE "*" .dfm"
TForm4 *Form4;
//-----
__fastcall TForm4::TForm4(TComponEnt* OwnEr)
    : TForm(OwnEr)
{
}
//-----
void __fastcall TForm4::Button4Click(TObjEct *SEndEr)
{
    if (RadioButtOn1->ChEckEd == truE)
    {
        Form1->ADOQuEry3->ActivE = "falsE";
        Form1->ADOQuEry3->SQL->CIEar();
        Form1->ADOQuEry3->SQL->TExt = "SELECT *FROM Авторизация
WHERE Фамилия LIKE '" + Form4->Edit1->TExt + "'" + " and Пароль LIKE '"
+ Form4->Edit2->TExt + "'" + " and Доступ LIKE '" + Form4->RadioButtOn1-
>Caption + "'";
        Form1->ADOQuEry3->ActivE = "truE";
        if ((Edit1->TExt == Form1->DBEdit16->TExt) && (Edit2->TExt ==
Form1->DBEdit17->TExt))
        {
            Form3->Show() ;
            Form4->ClosE();
            Form3->DBEdit10->VisiBlE = falsE;
            Form3->LaBE157->VisiBlE = falsE;
            Form3->DBEdit48->VisiBlE = falsE;
            Form3->LaBE158->VisiBlE = falsE;
            Edit1->CIEar();
            Edit2->CIEar();
        }
    }
    ElsE
    ShowMEssagE("Вы не ввели или не верно ввели логин или пароль");
}
if (RadioButtOn2->ChEckEd == truE)
{
    Form1->ADOQuEry3->ActivE = "falsE";
    Form1->ADOQuEry3->SQL->CIEar();
    Form1->ADOQuEry3->SQL->TExt = "SELECT *FROM Авторизация
WHERE Фамилия LIKE '" + Form4->Edit1->TExt + "'" + " and Пароль LIKE '"
```


Продолжение приложения Б

```
+ Form4->Edit2->TExt + "" + " and Доступ LIKE " + Form4->RadioButton2->Caption + "";  
    Form1->ADOQuery3->Active = "true";  
    if ((Edit1->TExt == Form1->DBEdit16->TExt) && (Edit2->TExt ==  
Form1->DBEdit17->TExt))  
    {  
        Form5->Show();  
        Form5->ADOQuery1->Active = false;  
        Form5->ADOQuery1->SQL->Clear();  
        Form5->ADOQuery1->SQL->Add("SElect      п.Фамилия,  
п.Имя, п.Отчество, б.К1_Р К1, б.К2_Р К2, б.К3_Р К3, б.К4_Р К4, б.К5_Р К5,  
б.К6_Р К6, б.К_Рейт Рейтинг, п.id_Препо, б.id_Бал, р.id_Рейтинг,  
р.id_Препо, р.id_Бал");  
        Form5->ADOQuery1->SQL->Add("FROM Баллы б, Рейтинг  
р, Преподаватели п");  
        Form5->ADOQuery1->SQL->Add("where      п.Фамилия  
Like('%"+Form1->DBEdit7->TExt+"%') and п.id_Препо = р.id_Препо and  
б.id_Бал = р.id_Бал ");  
        Form5->ADOQuery1->Active = true;  
        Form3->Visible = false;  
        Form4->Close();  
        Edit1->Clear();  
        Edit2->Clear();  
    }  
    Else  
    ShowMessage("Вы не ввели или не верно ввели логин или пароль");  
    }  
    if (RadioButton3->Checked == true)  
    {  
        Form1->ADOQuery3->Active = "false";  
        Form1->ADOQuery3->SQL->Clear();  
        Form1->ADOQuery3->SQL->TExt = "SELECT *FROM Авторизация  
WHERE Фамилия LIKE " + Form4->Edit1->TExt + "" + " and Пароль LIKE " +  
Form4->Edit2->TExt + "" + " and Доступ LIKE " + Form4->RadioButton3->  
Caption + "";  
        Form1->ADOQuery3->Active = "true";  
        if((Edit1->TExt == Form1->DBEdit16->TExt) && (Edit2->TExt ==  
Form1->DBEdit17->TExt))  
        {  
            Form6->Show();  
            Form6->ADOQuery1->Active = true;  
            Form4->Close();
```

Продолжение приложения Б

```
        Edit1->Clear();
        Edit2->Clear();
    }
    ElseE
    ShowMessage("Вы не ввели или не верно ввели логин или пароль");
}
}
//-----
```

```
void __fastcall TForm4::Button6Click(TObject *SEndEr)
{
    Form4->Close();
    Form1->Show();
    Edit1->Clear();
    Edit2->Clear();
}
//-----
```

```
void __fastcall TForm4::Button5Click(TObject *SEndEr)
{
    Form7->Show();
    Form4->Close();
    Edit1->Clear();
    Edit2->Clear();
}
}
```

Шестая форма

```
#include <vcl.h>
#pragma hdrstop
```

```
#include "Unit5.h"
#include "Unit4.h"
#include "Unit3.h"
#include "Unit1.h"
```

```
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm5 *Form5;
//-----
__fastcall TForm5::TForm5(TComponent* Owner)
    : TForm(Owner)
```

Продолжение приложения Б

```
{
}
//-----
void __fastcall TForm5::Button1Click(TObject *SEndEr)
{
Form1->Show();
Form5->Close();

}
//-----
void __fastcall TForm5::Button2Click(TObject *SEndEr)
{
Edit1->Text = FloatToStr(StrToFloat(DBEdit1-
>Text)+StrToFloat(DBEdit2->Text)+StrToFloat(DBEdit3-
>Text)+StrToFloat(DBEdit5->Text)+StrToFloat(DBEdit6->Text) +
StrToFloat(DBEdit7->Text));
DBEdit4->Show();
ShowMessage("Заполните окно - Введите итоговый балл");
}
//-----

void __fastcall TForm5::DBNavigator1BeforeAction(TObject *SEndEr,
TNavigatEBtn Button)

{
if (DBEdit4->Text != Edit1->Text) {
ShowMessage("Не верное значение или символ!");
DBNavigator1->Enabled = false;
Button1->Visible = false;
}
Else {
ShowMessage("Данные добавлены в базу!");
DBNavigator1->Enabled = true;
Button1->Visible = true;
}
}
}
```

Седьмая форма

```
#include <vcl.h>
#pragma hdrstop
```

Продолжение приложения Б

```
#include "Unit6.h"
#include "Unit1.h"
#include "Unit8.h"
#include "Unit10.h"
//-----
#pragma packagE(smart_init)
#pragma rEsourcE "*.dfm"
TForm6 *Form6;
//-----
__fastcall TForm6::TForm6(TComponEnt* OwnEr)
    : TForm(OwnEr)
{
}
//-----
void __fastcall TForm6::Button1Click(TObjEct *SEndEr)
{
    ADOQuEry1->ActivE = falsE;
    Form1->Show();
    Form6->ClosE();
}
//-----
void __fastcall TForm6::Button2Click(TObjEct *SEndEr)
{
    Form8->Show();
    Form6->ClosE();
}
//-----

void __fastcall TForm6::Button3Click(TObjEct *SEndEr)
{
    Form10->Show();
    Form6->ClosE();
}
Восьмая форма

#include <vcl.h>
#pragma hdrstop

#include "Unit7.h"
#include "Unit1.h"
#include "Unit4.h"
```

Продолжение приложения Б

```
//-----  
#pragma packagE(smart_init)  
#pragma rEsourcE "*.dfm"  
TForm7 *Form7;  
//-----  
__fastcall TForm7::TForm7(TComponEnt* OwnEr)  
    : TForm(OwnEr)  
{  
}  
//-----  
void __fastcall TForm7::Button1Click(TObjEct *SEndEr)  
{  
    Form7->ClosE();  
    Form4->Show();  
}
```

Девятая форма

```
#includE <vcl.h>  
#pragma hdrstop  
  
#includE "Unit8.h"  
#includE "Unit1.h"  
#includE "Unit6.h"  
#includE "Unit9.h"  
//-----  
#pragma packagE(smart_init)  
#pragma rEsourcE "*.dfm"  
TForm8 *Form8;  
//-----  
__fastcall TForm8::TForm8(TComponEnt* OwnEr)  
    : TForm(OwnEr)  
{  
}  
//-----  
void __fastcall TForm8::Button5Click(TObjEct *SEndEr)  
{  
    Form8->ClosE();  
    Form6->Show();  
}  
//-----
```

Продолжение приложения Б

```
void __fastcall TForm8::Button3Click(TObject *SEndEr)
{
    Edit1->TExt = StrToInt(LaBE1EdEdit1->TExt) + StrToInt(LaBE1EdEdit2-
>TExt) + StrToInt(LaBE1EdEdit3->TExt) + StrToInt(LaBE1EdEdit4->TExt);
    DBEdit5->TExt = Edit1->TExt;
}
//-----

void __fastcall TForm8::Button1Click(TObject *SEndEr)
{
    Form8->Hide();
    Form9->Show();

}
//-----

void __fastcall TForm8::Button4Click(TObject *SEndEr)
{
    int a,B,c,d,E,f,g,h;
    a = StrToFloat(DBEdit2->TExt);
    B = StrToFloat(DBEdit3->TExt);
    c = StrToFloat(DBEdit4->TExt);
    d = StrToFloat(DBEdit5->TExt);
    E = StrToFloat(DBEdit6->TExt);
    f = StrToFloat(DBEdit7->TExt);
    g = StrToFloat(DBEdit8->TExt);
    h = StrToFloat(DBEdit9->TExt);
    Edit2->TExt = FloatToStr((a+B+c+d+E+f+h+g));
    DBEdit10->TExt == " ";
    DBEdit10->VisiBLE = true;
    LaBE110->VisiBLE = true;
    ShowMEssagE("Заполните окно - Введите итоговый балл");
}
//-----

void __fastcall TForm8::DBNavigator1BEforEAction(TObject *SEndEr,
TNavigatEBtn Button)

{
    if (DBEdit10->TExt != Edit2->TExt) {
        ShowMEssagE("Не верное значение или символ!");
        DBNavigator1->EnaBIEd = false;
    }
}
```

Продолжение приложения Б

```
        Button5->Visible = false;
    }
    Else {
        ShowMessage("Данные добавлены в базу!");
        DBNavigator1->Enabled = true;
        Button5->Visible = true;
    }
}
```

Десятая форма

```
#include <vcl.h>
#pragma hdrstop

#include "Unit9.h"
#include "Unit8.h"
#include "Unit1.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm9 *Form9;
//-----
__fastcall TForm9::TForm9(TComponent* Owner)
    : TForm(Owner)
{
}
//-----

void __fastcall TForm9::Button1Click(TObject *SEnder)
{
    Edit1->Clear();
    Edit2->Clear();
    Edit3->Clear();
    Edit4->Clear();
    Edit5->Clear();
    Edit6->Clear();
    Edit7->Clear();
    Edit8->Clear();
    Edit9->Clear();
    Edit10->Clear();
    Edit11->Clear();
    Edit12->Clear();
}
```

Продолжение приложения Б

Edit13->Clear();
Edit14->Clear();
Edit15->Clear();
Edit16->Clear();
Edit17->Clear();
Edit18->Clear();
Edit19->Clear();
Edit20->Clear();
Edit21->Clear();
Edit22->Clear();
Edit23->Clear();
Edit23->Clear();
Edit24->Clear();
Edit25->Clear();
Edit26->Clear();
Edit27->Clear();
Edit28->Clear();
Edit29->Clear();
Edit30->Clear();
Edit31->Clear();
Edit32->Clear();
Edit33->Clear();
Edit34->Clear();
Edit35->Clear();
Edit36->Clear();
Edit37->Clear();
Edit38->Clear();
Edit39->Clear();
Edit40->Clear();
Edit41->Clear();
Edit42->Clear();
Edit43->Clear();
Edit44->Clear();
Edit45->Clear();
Edit46->Clear();
Edit47->Clear();
Edit48->Clear();
Edit49->Clear();
Edit50->Clear();
Edit51->Clear();
Edit52->Clear();
Edit53->Clear();

Продолжение приложения Б

Edit54->Clear();
Edit55->Clear();
Edit56->Clear();
Edit57->Clear();
Edit58->Clear();
Edit59->Clear();
Edit60->Clear();
Edit61->Clear();
Edit62->Clear();
Edit63->Clear();
Edit64->Clear();
Edit65->Clear();
Edit66->Clear();
Edit67->Clear();
Edit68->Clear();
Edit69->Clear();
Edit70->Clear();
Edit71->Clear();
Edit72->Clear();
Edit73->Clear();
Edit74->Clear();
Edit75->Clear();
Edit76->Clear();
Edit77->Clear();
Edit78->Clear();
Edit79->Clear();
Edit80->Clear();
Edit81->Clear();
Edit82->Clear();
Edit83->Clear();
Edit84->Clear();
Edit85->Clear();
Edit86->Clear();
Edit87->Clear();
Edit88->Clear();
Edit89->Clear();
Edit90->Clear();
Edit91->Clear();
Edit92->Clear();
Edit93->Clear();
Edit94->Clear();
Form8->Show();

Продолжение приложения Б

```
Form9->ClosE();
}
//-----
void __fastcall TForm9::Button2Click(TObject *SEndEr)
{
    int a,B,c,d,E,f,g,h,r;
    if ((Edit1->TExt == "") || (Edit2->TExt == "") || (Edit3->TExt == "") ||
(Edit4->TExt == "") || (Edit5->TExt == "") || (Edit6->TExt == "") || (Edit7->TExt
== "") || (Edit8->TExt == "") || (Edit9->TExt == ""))
    {
        ShowMessage("Заполните данные");
    }
    Else {
        Edit10->TExt = IntToStr(StrToInt(DBEdit1->TExt) * StrToInt(Edit1-
>TExt));
        Edit11->TExt = IntToStr(StrToInt(DBEdit2->TExt) * StrToInt(Edit2-
>TExt));
        Edit12->TExt = IntToStr(StrToInt(DBEdit3->TExt) * StrToInt(Edit3-
>TExt));
        Edit13->TExt = IntToStr(StrToInt(DBEdit4->TExt) * StrToInt(Edit4-
>TExt));
        Edit14->TExt = IntToStr(StrToInt(DBEdit5->TExt) * StrToInt(Edit5-
>TExt));
        Edit15->TExt = IntToStr(StrToInt(DBEdit6->TExt) * StrToInt(Edit6-
>TExt));
        Edit16->TExt = IntToStr(StrToInt(DBEdit7->TExt) * StrToInt(Edit7-
>TExt));
        Edit17->TExt = IntToStr(StrToInt(DBEdit8->TExt) * StrToInt(Edit8-
>TExt));
        Edit18->TExt = IntToStr(StrToInt(DBEdit9->TExt) * StrToInt(Edit9-
>TExt));
        a = StrToInt(Edit10->TExt);
        B = StrToInt(Edit11->TExt);
        c = StrToInt(Edit12->TExt);
        d = StrToInt(Edit13->TExt);
        E = StrToInt(Edit14->TExt);
        f = StrToInt(Edit15->TExt);
        g = StrToInt(Edit16->TExt);
        h = StrToInt(Edit17->TExt);
        r = StrToInt(Edit18->TExt);
        Edit19->TExt = 1 + IntToStr((a+B+c+d+E+f+g-h-r)/20);
        DBEdit10->TExt == " ";
    }
}
```

Продолжение приложения Б

```
DBEdit10->Visible = true;
Label157->Visible = true;
ShowMessage("Заполните окно - Введите итоговый балл");
}
}
//-----

void __fastcall TForm9::Button4Click(TObject *SEnder)
{
int
q,w,E,r,t,y,u,i,o,p,a,s,d,f,g,h,j,k,l,z,x,c,v,B,n,m,qw,qE,qR,qT,qY,qU,qI,qO,qP,qA,qS,qD
;
    Edit57->Clear();
    // Проверка на пустоту Едитов
    if ((Edit20->Text == "") || (Edit21->Text == "") || (Edit22->Text == "") ||
(Edit23->Text == "") || (Edit24->Text == "") || (Edit25->Text == "") || (Edit26-
>Text == "") || (Edit27->Text == "") || (Edit28->Text == "") || (Edit29->Text ==
"") || (Edit30->Text == "") || (Edit31->Text == "") || (Edit43->Text == "") ||
(Edit32->Text == "") || (Edit33->Text == "") || (Edit34->Text == "") || (Edit35-
>Text == "") || (Edit36->Text == "") || (Edit37->Text == "") || (Edit38->Text ==
"") || (Edit39->Text == "") || (Edit40->Text == "") || (Edit41->Text == "") ||
(Edit42->Text == "") || (Edit55->Text == "") || (Edit44->Text == "") || (Edit45-
>Text == "") || (Edit46->Text == "") || (Edit47->Text == "") || (Edit56->Text ==
"") || (Edit48->Text == "") || (Edit49->Text == "") || (Edit50->Text == "") ||
(Edit51->Text == "") || (Edit52->Text == "") || (Edit53->Text == "") || (Edit54-
>Text == ""))
    {
        ShowMessage("Заполните данные");
    }
    else {
        //Первый столбец
        Edit58->Text = IntToStr(StrToInt(DBEdit11->Text) * StrToInt(Edit20-
>Text));
        Edit59->Text = IntToStr(StrToInt(DBEdit12->Text) * StrToInt(Edit21-
>Text));
        Edit60->Text = IntToStr(StrToInt(DBEdit13->Text) * StrToInt(Edit22-
>Text));
        Edit61->Text = IntToStr(StrToInt(DBEdit35->Text) * StrToInt(Edit23-
>Text));
        Edit62->Text = IntToStr(StrToInt(DBEdit36->Text) * StrToInt(Edit24-
>Text));
        Edit63->Text = IntToStr(StrToInt(DBEdit37->Text) * StrToInt(Edit25-
>Text));
    }
}
```

Продолжение приложения Б

```
    Edit64->TExt = IntToStr(StrToInt(DBEdit38->TExt) * StrToInt(Edit26-
>TExt));
    Edit65->TExt = IntToStr(StrToInt(DBEdit39->TExt) * StrToInt(Edit27-
>TExt));
    Edit66->TExt = IntToStr(StrToInt(DBEdit40->TExt) * StrToInt(Edit28-
>TExt));
    Edit67->TExt = IntToStr(StrToInt(DBEdit41->TExt) * StrToInt(Edit29-
>TExt));
    Edit68->TExt = IntToStr(StrToInt(DBEdit42->TExt) * StrToInt(Edit30-
>TExt));
    Edit69->TExt = IntToStr(StrToInt(DBEdit43->TExt) * StrToInt(Edit31-
>TExt));
    //Второй столбец
    Edit70->TExt = IntToStr(StrToInt(DBEdit44->TExt) * StrToInt(Edit43-
>TExt));
    Edit71->TExt = IntToStr(StrToInt(DBEdit45->TExt) * StrToInt(Edit32-
>TExt));
    Edit72->TExt = IntToStr(StrToInt(DBEdit46->TExt) * StrToInt(Edit33-
>TExt));
    Edit73->TExt = IntToStr(StrToInt(DBEdit14->TExt) * StrToInt(Edit34-
>TExt));
    Edit74->TExt = IntToStr(StrToInt(DBEdit15->TExt) * StrToInt(Edit35-
>TExt));
    Edit75->TExt = IntToStr(StrToInt(DBEdit16->TExt) * StrToInt(Edit36-
>TExt));
    Edit76->TExt = IntToStr(StrToInt(DBEdit17->TExt) * StrToInt(Edit37-
>TExt));
    Edit77->TExt = IntToStr(StrToInt(DBEdit18->TExt) * StrToInt(Edit38-
>TExt));
    Edit78->TExt = IntToStr(StrToInt(DBEdit19->TExt) * StrToInt(Edit39-
>TExt));
    Edit79->TExt = IntToStr(StrToInt(DBEdit20->TExt) * StrToInt(Edit40-
>TExt));
    Edit80->TExt = IntToStr(StrToInt(DBEdit21->TExt) * StrToInt(Edit41-
>TExt));
    Edit81->TExt = IntToStr(StrToInt(DBEdit22->TExt) * StrToInt(Edit42-
>TExt));
    //Третий столбец
    Edit82->TExt = IntToStr(StrToInt(DBEdit23->TExt) * StrToInt(Edit55-
>TExt));
    Edit83->TExt = IntToStr(StrToInt(DBEdit24->TExt) * StrToInt(Edit44-
>TExt));
```

Продолжение приложения Б

```
    Edit84->TExt = IntToStr(StrToInt(DBEdit25->TExt) * StrToInt(Edit45->TExt));
    Edit85->TExt = IntToStr(StrToInt(DBEdit26->TExt) * StrToInt(Edit46->TExt));
    Edit86->TExt = IntToStr(StrToInt(DBEdit27->TExt) * StrToInt(Edit47->TExt));
    Edit87->TExt = IntToStr(StrToInt(DBEdit47->TExt) * StrToInt(Edit56->TExt));
    Edit88->TExt = IntToStr(StrToInt(DBEdit28->TExt) * StrToInt(Edit48->TExt));
    Edit89->TExt = IntToStr(StrToInt(DBEdit29->TExt) * StrToInt(Edit49->TExt));
    Edit90->TExt = IntToStr(StrToInt(DBEdit30->TExt) * StrToInt(Edit50->TExt));
    Edit91->TExt = IntToStr(StrToInt(DBEdit31->TExt) * StrToInt(Edit51->TExt));
    Edit92->TExt = IntToStr(StrToInt(DBEdit32->TExt) * StrToInt(Edit52->TExt));
    Edit93->TExt = IntToStr(StrToInt(DBEdit33->TExt) * StrToInt(Edit53->TExt));
    Edit94->TExt = IntToStr(StrToInt(DBEdit34->TExt) * StrToInt(Edit54->TExt));
    // Присвоение значений Едитов буквам
    q = StrToInt(Edit58->TExt) + StrToInt(Edit59->TExt) + StrToInt(Edit60->TExt) + StrToInt(Edit61->TExt) + StrToInt(Edit62->TExt) + StrToInt(Edit63->TExt) + StrToInt(Edit64->TExt) + StrToInt(Edit65->TExt);
    o = StrToInt(Edit66->TExt) + StrToInt(Edit67->TExt) + StrToInt(Edit68->TExt) + StrToInt(Edit69->TExt) + StrToInt(Edit70->TExt) + StrToInt(Edit71->TExt) + StrToInt(Edit72->TExt) + StrToInt(Edit73->TExt);
    j = StrToInt(Edit74->TExt) + StrToInt(Edit75->TExt) + StrToInt(Edit76->TExt) + StrToInt(Edit77->TExt) + StrToInt(Edit78->TExt) + StrToInt(Edit79->TExt) + StrToInt(Edit80->TExt) + StrToInt(Edit81->TExt);
    m = StrToInt(Edit82->TExt) + StrToInt(Edit83->TExt) + StrToInt(Edit84->TExt) + StrToInt(Edit85->TExt) + StrToInt(Edit86->TExt) + StrToInt(Edit87->TExt) + StrToInt(Edit88->TExt) + StrToInt(Edit89->TExt);
    qo= StrToInt(Edit90->TExt) + StrToInt(Edit91->TExt) + StrToInt(Edit92->TExt) + StrToInt(Edit93->TExt) + StrToInt(Edit94->TExt);
    // Итоговый подсчет
    Edit57->TExt = IntToStr(q + o + j + m + qo)/20;
    DBEdit48->TExt == " ";
    DBEdit48->VisiBIE = truE;
    LaBE158->VisiBIE = truE;
```

Продолжение приложения Б

```
ShowMessage("Заполните окно - Введите итоговый балл");
}
}
//-----

void __fastcall TForm9::DBNavigator1BEforEAction(TObject *SEndEr,
TNavigatorBtn Button)

{
if (DBEdit10->Text != Edit19->Text) {
    ShowMessage("Не верное значение или символ!");
    DBNavigator1->Enabled = false;
    Button2->Visible = false;
    Button3->Visible = false;
}
else {
    ShowMessage("Данные добавлены в базу!");
    DBNavigator1->Enabled = true;
    Button2->Visible = true;
    Button3->Visible = true;
}
}
//-----

void __fastcall TForm9::DBNavigator2BEforEAction(TObject *SEndEr,
TNavigatorBtn Button)

{
if (DBEdit48->Text != Edit57->Text) {
    ShowMessage("Не верное значение или символ!");
    DBNavigator2->Enabled = false;
    Button2->Visible = false;
    Button3->Visible = false;
}
else {
    ShowMessage("Данные добавлены в базу!");
    DBNavigator2->Enabled = true;
    Button2->Visible = true;
    Button3->Visible = true;
}
}
}
```

Продолжение приложения Б

```
//-----  
void __fastcall TForm9::DBEdit10Change(TObject *SEndEr)  
{  
  DBNavigator1->Enabled = true;  
}  
//-----  
  
void __fastcall TForm9::DBEdit48Change(TObject *SEndEr)  
{  
  DBNavigator2->Enabled = true;  
}  
//-----  
  
void __fastcall TForm9::Button3Click(TObject *SEndEr)  
{  
  Edit95->Text = FloatToStr(StrToFloat(DBEdit49->Text) +  
  StrToFloat(DBEdit50->Text));  
  DBEdit51->Text == " ";  
  DBEdit51->Visible = true;  
  Label62->Visible = true;  
  ShowMessage("Заполните окно - Введите итоговый балл");  
}  
//-----  
  
void __fastcall TForm9::DBNavigator3BeforeAction(TObject *SEndEr,  
TNavigatEBtn Button)  
  
{  
  if (DBEdit51->Text != Edit95->Text) {  
    ShowMessage("Не верное значение или символ!");  
    DBNavigator3->Enabled = false;  
    Button1->Visible = false;  
  }  
  Else {  
    ShowMessage("Данные добавлены в базу!");  
    DBNavigator3->Enabled = true;  
    Button1->Visible = true;  
  }  
}
```