

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

кафедра Компьютерные технологии

«Допущен к защите»
Заведующий кафедрой Кураббаев З.К. доцент, проф.
(Ф.И.О., ученая степень, звание)
« 21 » мая 2014 г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: «Защита БДЗЫ ДАННЫХ SQL при помощи Microsoft Access на примере БДЗЫ ДАННЫХ «РЕГИСТР ПРЕПОДАВАТЕЛЕЙ»

Специальность 5В070400 - Вычислительная техника и программное обеспечение

Выполнил (а) Куанышев АЕ ВТ-10-05
(Фамилия и инициалы) группа

Научный руководитель Сатпаева Е.Г., к.т.н., доцент
(Фамилия и инициалы, ученая степень, звание)

Консультанты:

по экономической части:

Зрешеева З.Д., с.г. преподаватель
(Фамилия и инициалы, ученая степень, звание)
Зрешеева « 13 » 05 2014 г.
(подпись)

по безопасности жизнедеятельности:

Дригалева И.Г., д.т.н., профессор
(Фамилия и инициалы, ученая степень, звание)
Дригалева « 04 » 05 2014 г.
(подпись)

по применению вычислительной техники:

Сатпаева Е.Г., к.т.н., доцент
(Фамилия и инициалы, ученая степень, звание)
« _____ » _____ 20 _____ г.
(подпись)

(Фамилия и инициалы, ученая степень, звание)

« _____ » _____ 20 _____ г.

(подпись)

Нормоконтролер: Тусупов Д.М.
(Фамилия и инициалы, ученая степень, звание)

Тусупов « 20 » мая 2014 г.
(подпись)

Рецензент: _____
(Фамилия и инициалы, ученая степень, звание)

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Факультет „Информационные технологии“
Специальность Вычислительная техника и программное обеспечение
Кафедра „Компьютерные технологии“

ЗАДАНИЕ

на выполнение дипломного проекта

Студент Куанышев Аманжол Ермакұлы
(фамилия, имя, отчество)
Тема проекта Защита базы данных SQL при помощи Microsoft Azure на примере базы данных - рейтинг преподавателей

утверждена приказом ректора № 115 от «24» сентября 2013 г.
Срок сдачи законченной работы «3» июня 2014 г.
Исходные данные к проекту требуемые параметры результатов проектирования (исследования) и исходные данные объекта
информация по проектированию базы данных, документация Microsoft Azure

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

- Анализ защиты данных в Microsoft Azure
- Создание логической модели базы данных
- Концептуальное проектирование:
- Технико-экономические обоснования
- безопасности информации

Перечень графического материала (с точным указанием обязательных чертежей)

| |
|--|
| Авторизация пользователя в Microsoft Azure |
| Диаграмма физической модели БАЗЫ ДАННЫХ |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |

Рекомендуемая основная литература

| |
|--|
| Сатинова Е.Г. Проектирование БДЗ ДАННЫХ: МЕТОДИЧЕСКИЕ УКАЗАНИЯ к выполнению лабораторных работ - Алматы: АИЭС, 2009. |
| Microsoft SQL Server 2008. Реализация и программирование, - М.: «Русская Редакция», СПб: «Питер», 2007. |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |
| |


Консультанты по проекту с указанием относящихся к ним разделов


| Раздел | Консультант | Сроки | Подпись |
|---------------|---------------|------------------|----------------------|
| БДЗ | Дружкова И.Г. | 11.04 - 04.05.14 | <i>И.Г. Дружкова</i> |
| Модель БД | Бреснева З.Д. | 02.04 - 13.08.14 | <i>З.Д. Бреснева</i> |
| Нормоконтроль | Туснов Д.М. | 20.05.14 | <i>Д.М. Туснов</i> |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Г Р А Ф И К
подготовки дипломного проекта

| № п/п | Наименование разделов, перечень разрабатываемых вопросов | Сроки представления руководителю | Примечание |
|----------|---|--|------------|
| | Анализ защиты данных в Azure | 09.04.14 | |
| | Создание модели модели БД | 10.04.14 | |
| | Оцен начальной разработки БД | 17.04.14 | |
| | Концептуальное проектирование | 24.04.14 | |
| | Технико-экономическое обоснование | 13.05.14 | |
| | Безопасность информации | 04.05.14 | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

Дата выдачи задания « 3 » марта 2014 г.

Заведующий кафедрой _____
(подпись)  Курамбаев Б.К.
(Фамилия и инициалы)

Руководитель _____
(подпись)  Сатимова Е.Г.
(Фамилия и инициалы)

Задание принял к исполнению _____
 Кознетсов А.Е.

Бұл бітіру жұмысында тапсырмаға сәйкес кафедра меңгерушісі мен оқытушылардың рейтингін есептеуді елеулі жеңілдететін және автоматтандыруға мүмкіндік беретін бағдарламалық кешенді құру жүзеге асырылды. Қосымшалардың әртүрлі шарттарға бейімділігін қамтамасыз ету үшін бұл бағдарламалардың әртүрлі компьютерлерде бірдей жұмыс істеуіне мүмкіндік беретін технологияларға таңдау жасалған. Сонымен бірге мәліметтер базасымен ыңғайлы жұмыс істеуді қамтамасыз ететін алгоритмдер құрылған. Мәліметтер базасы мен қосымшалар КТ кафедрасында сыналды. Берілген бағдарламалық кешен университеттің кез-келген кафедрасына бейімделе алады. Ғылыми-зерттеу жұмысына шыққан шығындар есептелді. Өміртіршілік қауіпсіздігі мәселелері, жұмыс шарттары қарастырылды, жұмыс орындарының жарықтандырылуына және ауа баптауларына есептеулер жүргізілді.

АННОТАЦИЯ

В данной выпускной работе согласно заданию была осуществлена разработка программного комплекса, позволяющего автоматизировать и значительно упростить расчет рейтинга преподавателей и заведующих кафедрами. Для обеспечения адаптивности приложения к различным условиям осуществлен выбор технологий, позволяющих этим программам одинаково функционировать на различных компьютерах, а также разработаны алгоритмы, которые обеспечивают удобную систему работы с базой данных. База данных и приложение были протестированы на кафедре КТ. Данный программный комплекс может быть адаптирован на любой кафедре университета. Произведен расчет затрат на научно-исследовательскую работу. Рассмотрены вопросы безопасности жизнедеятельности, а именно расчет кондиционирования помещения и освещенности.

ANNOTATION

In this graduating project, according to the task was carried the development of program complex allowing to automate and simplify calculating of instructors rating and head of department. For ensure adaptability of applications to various conditions was implemented the choice of technology who allows to these programs equally operate on the different computers. Also algorithms was developed who ensure convenient system of database work. Database and application was tested in the chair of Computers Technology. This software can be adapted to any department of the university. Also the cost of research work was calculated. The issues of life safety, namely the air-conditioning and room lighting was considered.

| | | |
|---|--|----|
| | Введение | 8 |
| 1 | Определение облачных серверов | 9 |
| | 1.1 Понятие Microsoft Windows Azure | 9 |
| | 1.2 Криптографические сервисы и защита данных в Windows Azure | 10 |
| | 1.2.1 Хранилище ключей | 11 |
| | 1.2.2 Шифрование в SQL Azure | 12 |
| | 1.3 Шифрование в SQL Azure | 13 |
| | 1.3.1 Обзор платформы | 14 |
| | 1.3.2 Сервисы | 14 |
| 2 | Разработка базы данных | 15 |
| | 2.1 Анализ предметной области | 15 |
| | 2.1.1 Коэффициент K1 | 15 |
| | 2.1.2 Коэффициент K2 | 15 |
| | 2.1.3 Коэффициент K3 | 16 |
| | 2.1.4 Коэффициент K4 | 16 |
| | 2.1.5 Коэффициент K5 | 16 |
| | 2.1.6 Коэффициент K6 | 16 |
| | 2.1.7 Коэффициент K7 | 17 |
| | 2.2 UML диаграммы | 18 |
| | 2.2.1 Диаграмма компонентов | 18 |
| | 2.2.2 Диаграмма прецедентов | 19 |
| | 2.2.3 Диаграмма классов | 19 |
| | 2.2.4 Диаграмма развертывания | 20 |
| 3 | Проектирование БД | 21 |
| | 3.1 Концептуальное проектирование | 21 |
| | 3.2 Определение атрибутов каждой сущности | 24 |
| | 3.3 Составление реляционных отношений | 27 |
| | 3.4. Расчет места для хранения БД | 29 |
| 4 | Логическое проектирование | 35 |
| | 4.1 Схемы отношений составленные на языке определения данных (ddl, datadefinitionlanguage) | 35 |
| | 4.2 Заполнение созданной БД | 38 |
| | 4.2.1 Заполнение таблицы Виды анкетирования | 38 |
| | 4.2.2 Заполнение таблицы Авторство | 39 |
| | 4.2.3 Заполнение таблицы Деканат | 39 |
| | 4.2.4 Заполнение таблицы Физ. воспитания | 39 |
| | 4.2.5 Заполнение таблицы Оценки деканата | 39 |
| | 4.2.6 Заполнение таблицы Работа | 40 |
| | 4.2.7 Заполнение таблицы Виды работ | 40 |
| | 4.2.8 Заполнение таблицы Преподаватель | 41 |
| | 4.3. Запросы базы данных. Представления, процедуры и триггеры | 43 |
| | 4.3.1 Представления | 43 |

| | | |
|-------|---|----|
| 4.3.2 | Процедуры | 45 |
| 4.3.3 | Триггеры | 47 |
| 4.3.4 | Функции | 52 |
| 5 | Описание интерфейса Windows Azure. Работа с приложением | 54 |
| 6 | Безопасность жизнедеятельности | 59 |
| 6.1 | Анализ потенциально опасных и вредных факторов, влияющих на работу программиста | 59 |
| 6.2. | Расчетная часть | 63 |
| 6.2.1 | Расчет уровня шума от персонального компьютера | 64 |
| 6.2.2 | Анализ данных при выборе монитора | 65 |
| 7 | Технико–экономическое обоснование | 70 |
| 7.1 | Расчет затрат на разработку информационных технологий | 70 |
| 7.2 | Расчет цены программного продукта | 79 |
| | Заключение | 81 |
| | Список литературы | 82 |
| | Приложение А | 83 |
| | Приложение Б | 84 |
| | Приложение В | 86 |
| | Приложение Г | 88 |
| | Приложение Д | 89 |

Введение

В наше время компьютерная техника используется повсеместно. Хранение и обработка данных является ключом к автоматизации и управлению многих трудовых процессов. Именно поэтому в их состав в качестве основного звена включается система управления базами данных. Но самая ответственная стадия разработки базы данных является ее защита.

Сегодня многие организации сталкиваются с проблемами защиты информации в базах данных. Одним из самых безопасных методов хранения данных является облачная технология. Самым надежным облачным сервером является Microsoft Windows Azure.

В дипломном проекте рассмотрена защита базы данных при помощи Windows Azure. Как пример была взята база данных рейтинга преподавателей.

Сначала будет описана защита базы данных при помощи Windows Azure. Будут подробно описаны методы шифрования, методы использования ключей.

После будут спроектированы такие объекты базы данных, как таблицы, представления, триггеры, функции, хранимые процедуры. Их взаимодействие будет описано в логической модели.

Рассмотрены вопросы безопасности жизнедеятельности и произведено технико-экономическое обоснование работы.

1 Определение облачных серверов

1.1 Понятие Microsoft Windows Azure

Microsoft (Windows) Azure – название облачной платформы от Microsoft. Платформа Windows Azure предоставляет возможность разработки и выполнения приложений и хранения данных на серверах, расположенных в распределенных дата-центрах. В 2014 году платформа была переименована в Microsoft Azure.

Microsoft Azure полностью реализует две облачные модели – платформы как сервиса (Platform as a Service, PaaS) и инфраструктуры как сервиса (Infrastructure as a Service, IaaS). Работоспособность платформы Windows Azure обеспечивает сеть глобальных дата-центров Microsoft.

Основные особенности данной модели:

- а) оплата только потребленных ресурсов;
- б) общая, многопоточная структура вычислений;
- в) абстракция от инфраструктуры.

В основе работы Microsoft Azure лежит запуск виртуальной машины для каждого экземпляра приложения. Разработчик определяет необходимый объем для хранения данных и требуемые вычислительные мощности (количество виртуальных машин), после чего платформа предоставляет соответствующие ресурсы. Когда первоначальные потребности в ресурсах изменяются, в соответствии с новым запросом заказчика платформа выделяет под приложение дополнительные или сокращает неиспользуемые ресурсы дата-центра.

Microsoft Azure как PaaS обеспечит не только все базовые функции операционной системы, но и дополнительные: выделение ресурсов по требованию для неограниченного масштабирования, автоматическую синхронную репликацию данных для повышения отказоустойчивости, обработку отказов инфраструктуры для обеспечения постоянной доступности и многое другое [1].

Microsoft Azure также реализует другой тип сервиса – инфраструктуру как сервис. Модель предоставления инфраструктуры (аппаратных ресурсов) реализует возможность аренды таких ресурсов, как серверы, устройства хранения данных и сетевое оборудование. Управление всей инфраструктурой осуществляется поставщиком, потребитель управляет только операционной системой и установленными приложениями. Такие сервисы оплачиваются по фактическому использованию и позволяют увеличивать или уменьшать объем инфраструктуры через специальный портал, предоставляемый поставщиками. В данной сервисной модели могут быть запущены практически любые приложения, установленные на стандартные образы ОС.

В доступной сразу галерее образов доступны образы следующих операционных систем: Windows Server, OpenSUSE, CentOS, Ubuntu, SUSE.

В 2013 году было представлено новое хранилище образцов виртуальных машин – VM Depot. VM Depot – это проект для сообщества Windows Azure,

запущенный командой Microsoft OpenTechnologies, Inc, ответственной за открытые технологии внутри Microsoft. Содержимое портала, а также настроенные для разных задач виртуальные машины, будут создаваться и публиковаться силами сообщества.

Microsoft Azure состоит из:

a) *Compute* – компонент, реализующий вычисления на платформе Windows Azure.

b) *Storage* – компонент хранилища предоставляет масштабируемое хранилище. Хранилища не имеет возможности использовать реляционную модель и является альтернативной, "облачной" версией SQL Server.

c) *Fabric* – Windows Azure Fabric по своему назначению является «контролёром» и ядром платформы, выполняя функции мониторинга в реальном времени, обеспечения отказоустойчивости, выделения мощностей, развертывания серверов, виртуальных машин и приложений, балансировки нагрузки и управления оборудованием.

Практически все сервисы Microsoft Azure имеют API, построенное на REST, что позволяет разработчикам использовать «облачные» сервисы с любой операционной системы, устройства и платформы [1].

Microsoft Azure была признана Compuware самой быстрой «облачной» платформой. Также бенчмарк Microsoft Azure показал высокую производительность для масштабных вычислений с результатами в 151,3 ТФлопс на 8064 ядрах с 90,2 процентной эффективностью. Аналитическая компания Nasuni представила очередное исследование провайдеров облачных сервисов хранения данных. Согласно этому отчету, платформа Microsoft Azure является лидером в тестах производительности при записи и чтении данных из облака, доступности данных и минимальному числу ошибок (0 %). В 2014 году Nasuni выпускает очередной отчет с результатами тестирования облачных хранилищ Amazon, Google, HP, Microsoft и Rackspace.

1.2 Криптографические сервисы и защита данных в Windows Azure

Windows Azure SDK расширяет базовые библиотеки .NET, чтобы разработчики могли интегрировать и использовать сервисы, предоставляемые Windows Azure. Доступ к CSP в проектах и сервисах Windows Azure не ограничен. То есть с учетом сборок, к которым уже привыкли, большая часть работы, относящейся к шифрованию и дешифрованию данных, останется прежней. Однако в нижележащей архитектуре произошли изменения, связанные с тем, когда шифруют данные и хранятся ключи [1].

1.2.1 Хранилище ключей

Как и при любой стратегии шифрования на уровне приложения или предприятия, инфраструктура шифрования и дешифрования – это лишь полдела. Настоящая проблема заключается в хранилище ключей и длительности их хранения. Надежность защиты зашифрованных данных определяется используемыми ключами, а эта проблема куда сложнее, чем можно подумать поначалу [2].

Когда дело доходит до хранения ключей в облачной среде, возникает важный вопрос: а где же держать ключи? Некоторые выражают озабоченность тем, что, храня ключи в облаке, подвергаешь свою защиту рискам, исходящим из самого облака. То есть некто может получить физический доступ к вашим данным, которые по умолчанию хранятся на диске в незашифрованном виде (как в случае Windows Azure).

Первое, что нужно сразу же отметить, – ни в каком приложении нельзя использовать какие-либо ключи, предоставляемые Windows Azure, в качестве шифровальных. Пример – ключи, предоставляемые Windows Azure для сервиса хранилища. Эти ключи конфигурируются так, чтобы их можно было легко менять для большей безопасности или на случай их компрометации. Другими словами – никто не гарантирует, что они и в будущем будут существовать [2].

Собственная библиотека ключей в Windows Azure Storage – хороший способ сохранить некую информацию в секрете, поскольку можно полагаться на эти данные как на надежные в многопользовательской среде и защищать их с помощью своих ключей хранилища. Это отличается от использования ключей хранилища в качестве шифровальных. Вместо этого можно применять ключи сервиса хранилища для доступа к библиотеке ключей, как делали бы это в случае любого другого хранимого файла. Реализовать такой вариант довольно легко. Создание ключей показано на рисунке 1.2

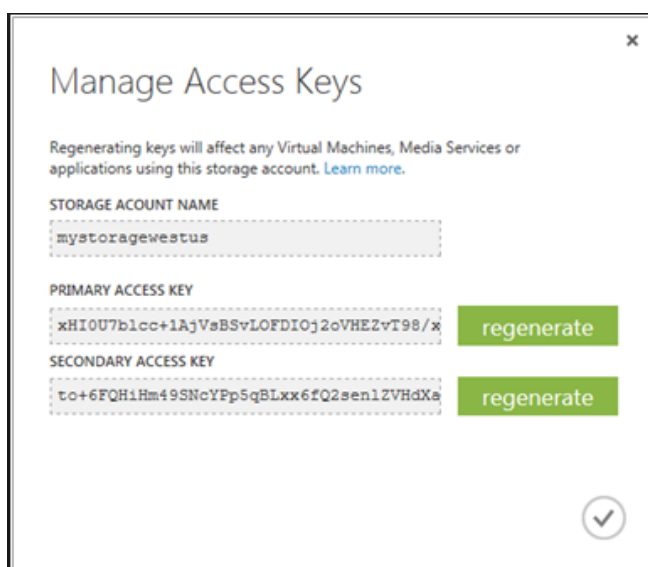


Рисунок 1.2 – Создание ключей в Microsoft Azure
1.2.2 Шифрование в SQL Azure

В SQL Server 2008 было введено новое средство: прозрачное шифрование данных (transparent data encryption, TDE). Впервые в SQL Server стало возможным полностью зашифровать данные с минимальными усилиями с вашей стороны по сравнению с тем, что требовалось для ограниченного шифрования, которое поддерживалось в SQL Server 2005. Однако начальная версия хранилища SQL Azure пока не поддерживает шифрование на уровне базы данных, хотя такая поддержка скорее всего будет включена в следующую версию. Следует заметить, что в настоящее время SQL Azure доступен только через TCP-соединения и только через порт 1433 [2].

Хотя это средство пока не интегрировано в Windows Azure, есть несколько других функций защиты SQL Azure, о которых должны знать разработчики или проектировщики. Во-первых, SQL Azure поддерживает поток табличных данных (tabular data stream, TDS). То есть по большей части можно подключаться к базе данных и взаимодействовать с ней точно так же, как всегда делали это. Об использовании преимуществ шифрования ADO.NET и сертификатов доверенных серверов определенно стоит подумать, особенно если вы обращаетесь к базе данных SQL Azure извне облака.

Свойства соединения Encrypt = True и Trust Server Certificate = False в должной комбинации обеспечат защиту передаваемых данных и помогут предотвратить атаки с посредниками (man-in-the-middle attacks). Кроме того, это обязательно для подключения к SQL Azure – невозможно соединиться с SQL Azure, пока не включите шифрование уровня соединения.

Второе средство защиты SQL Azure, с которым нужно ознакомиться, – брандмауэр SQL Azure. Он хорошо известен тем, кто пользовался локальными брандмауэрами или даже набором средств защиты SQL Server. Он разрешает или запрещает соединения из различных источников вплоть до конкретных IP-адресов или диапазонов. Брандмауэром SQL Azure можно управлять через портал SQL Azure или напрямую в главной базе данных с помощью хранимых процедур, таких как sp_set_firewall_rule и sp_delete_firewall_rule [2]. Добавление IP-адреса для брандмауэра показано на картинке 1.1

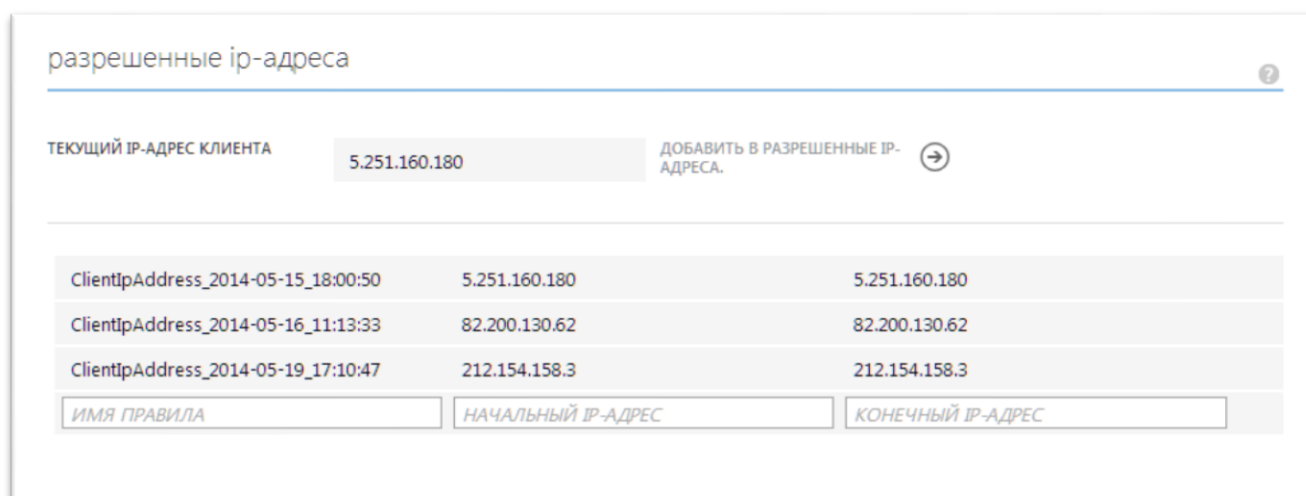


Рисунок 1.1 – Добавление IP-адреса

Как и в любой реализации SQL Server, управление пользовательскими учетными записями – еще один аспект, который нужно жестко контролировать. Брандмауэр в SQL Azure – действительно отличное средство, но полагаться только на него не стоит. Вы должны применять стойкие пароли в учетных записях и настраивать для них строго определенные права доступа, а также продумать свою модель защиты данных.

Благодаря этим новым средствам SQL Azure является высокозащищенной управляемой платформой для приложений в облаке. Если вы впервые пробуете работать с этим сервисом, не забудьте, что до попыток подключения вы должны сначала настроить брандмауэр SQL Azure. В первый раз это делается только через веб-портал SQL Azure, но впоследствии им можно управлять с помощью главной базы данных, как уже было сказано.

1.3 Описание SQL Azure

Windows Azure SQL Databases (первоначально SQL Server Data Services, позже SQL Services, позже Windows Azure SQL Databases) – это облачный сервис от корпорации Microsoft, предоставляющий возможность хранения и обработки реляционных данных, а также генерации отчетности. Предоставляет функциональность для различных сценариев синхронизации данных (локальная инфраструктура \Leftrightarrow облако, облако \Leftrightarrow облако). Является частью Windows Azure.

Windows Azure SQL Databases основан на Microsoft SQL Server, но предоставляет только подмножество типов данных. Поддерживаются основные типы: точные и приближенные числа, символьные строки (в том числе Юникод), дата и время, пространственные, двоичные и прочие типы данных.

Используется основанный на XML формат для передачи данных. Так же как и Microsoft SQL Server, Windows Azure SQL Databases использует T-SQL в качестве языка запросов. Tabular Data Stream (TDS) используется в качестве протокола для доступа к сервису через Интернет. По протоколу HTTP REST доступ не предоставляется. Microsoft рекомендует использовать ADO.NET Data Services для передачи данных и создания сервисов [3].

Пользователь может посылать Transact SQL запросы по протоколу TDS к сервису Windows Azure SQL Databases, и это позволяет приложениям использовать Windows Azure SQL Databases также, как они используют локальный SQL Server. Однако, поскольку Windows Azure SQL Databases является сервисом, его администрирование имеет свои особенности. В отличие от администрирования локального SQL Server, Windows Azure SQL Databases разделяет логический и физический аспекты администрирования. Клиент продолжает администрировать БД, управлять логинами, пользователями и

ролями, однако об оборудовании заботится Microsoft. В результате, Windows Azure SQL Databases предоставляет масштабируемый многопользовательский сервис баз данных с высочайшей степенью доступности, расширяемости, безопасности и самовосстановления

1.3.1 Обзор платформы

Платформа Windows Azure является «облачной» платформой для приложений, позволяющей хранить данные и выполнять приложения в дата центрах Microsoft. Windows Azure предоставляет «облачную» операционную систему, на основе которой работают все сервисы Azure и разработанные приложения. Платформа предлагает доступ к возможностям публичного облака. Используя публичное облако, клиент оплачивает только ресурсы и мощности, которые задействованы в приложение и только за фактическое время использования этих ресурсов. Основные особенности данной модели:

- а) оплата только потребленных ресурсов;
- б) общая, многопоточная структура вычислений;
- с) абстракция от инфраструктуры.

Работоспособность платформы Windows Azure обеспечивают 8 глобальных дата центров Microsoft.

На рисунке 1.1 показана среда разработки Windows Azure.

1.3.2 Сервисы

В остальных аспектах Microsoft Windows Azure SQL Databases значительно расширяет возможности SQL Server. В состав Windows Azure SQL Databases входят:

2 Разработка базы данных

2.1 Анализ предметной области

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

Администрация института устанавливает надбавку к должностному окладу преподавателя за счет собственных средств института по результатам его работы.

Надбавка к должностному окладу профессорско–преподавательского состава зависит от комплексного коэффициента (рейтинга) К, определяемого

для каждого преподавателя. Этот коэффициент включает несколько составляющих коэффициентов (K_1, K_2, \dots, K_7).

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

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

Комплексный коэффициент K определяется из следующего выражения

$$K=K_1+K_2+K_3+K_4+K_5+K_6+K_7 \quad (2.1)$$

Составляющие $K_1, K_2 \dots$ и т.д. определяются с учетом приведенной ниже таблицы.

2.1.1 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.

$$K_1 = 1 + \frac{\text{Сумма баллов (таблица 1.1)}}{20} \quad (2.2)$$

2.1.2 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 УМО. Вычисления K_2 проводит УМО.

$$K_2 = \frac{\text{Сумма баллов (таблица 1.2)}}{20} \quad (2.3)$$

2.1.3 Коэффициент K_3

K_3 учитывает оценку труда преподавателя в области науки. Вычисления проводятся самим преподавателем и утверждаются начальником НИС.

Коэффициент обновляется в течение семестра. Выполненные работы учитываются в течение года, т.е. при проведении двух рейтинговых оценок.

$$K3 = 1 + \frac{\text{Сумма баллов(таблица 1.3)}}{20} \quad (2.4)$$

2.1.4 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.

2.1.5 Коэффициент K5

K5 учитывает оценку труда преподавателя студентами. K5 равен среднему баллу 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.

2.1.6 Коэффициент K6

K6 учитывает оценку труда преподавателя 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, исходя из приложения таблицы 3.

$$K6 = \frac{\text{Баллы(таблица 1.4)}}{20} \quad (2.5)$$

2.1.7 Коэффициент K7

K7 учитывает оценку труда преподавателя института в качестве куратора студенческой группы и устанавливается деканом факультета. Диапазон изменений коэффициента K7 устанавливается от 0 до 2 с шагом 0,5. K7 учитывает работу преподавателя в предыдущем семестре.

Предварительная модель базы данных представлены на рисунке 2.1. Описание сущностей показано в таблице 2.1.

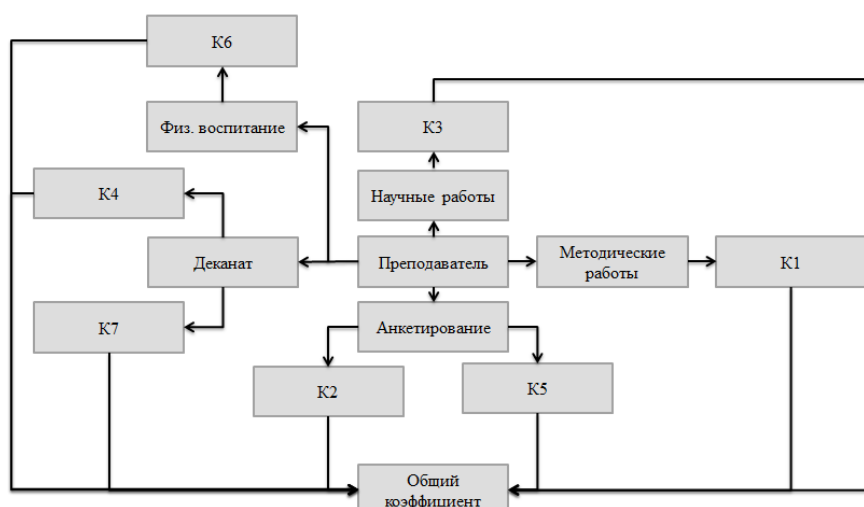


Рисунок 2.1 – Предварительная модель рейтинга преподавателей

Т а б л и ц а 2.1 – Описание сущностей

| Имя сущности | Описание сущности | Тип сущностей |
|-----------------|--|---------------|
| Преподаватель | Данные о преподавателях | Слабая |
| Кафедра | Данные о кафедрах | Сильная |
| Авторство | Данные об авторстве | Слабая |
| Деканат | Данные о деканатах | Сильная |
| Физ. Воспитание | Данные о физическом воспитание студентов | Слабая |
| Коэффициент | Данные о коэффициентах | Слабая |
| Оценка | Данные об оценка, выставляемых деканатом | Слабая |

Окончание таблицы 2.1

| Имя сущности | Описание сущности | Тип сущностей |
|---------------------|---|---------------|
| Работа | Данные о работах преподавателей | Слабая |
| Вид | Данные о видах работ | Сильная |
| Вид физ. воспитания | Данные о видах физического воспитания студентов | Сильная |
| Вид оценки | Данные о видах оценок, выставляемых деканатом | Слабая |
| Анкетирование | Данные об анкетирование преподавателей | Слабая |
| Методические работы | Данные о методических работах преподавателей | Слабая |

2.2 UML диаграммы

Для моделирования статических объектов в объектно–ориентированной концепции UML существует 4 типа диаграмм:

- 1 Диаграмма компонентов.
- 2 Диаграмма прецедентов.
- 3 Диаграмма классов.
- 4 Диаграмма развертывания.

2.2.1 Диаграмма компонентов

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

В данном случае исполняемый модуль Figure_Skating.exe обращается за получением или добавлением информации к базе данных Figure_.mdf. Но обращается не напрямую, а через SQLSERVER. Таким образом SQLSERVER является интерфейсом между программой и базой данных, обеспечивая их взаимодействие. Диаграмма компонентов представлена на рисунке 2.2.

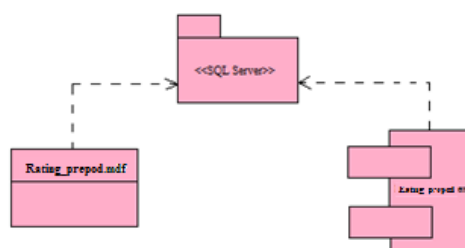


Рисунок 2.2 – Диаграмма компонентов.

2.2.2 Диаграмма прецедентов

Для характеристики взаимодействия пользователей с базами данных применяется диаграмма прецедентов или вариантов использования. Диаграмма прецедентов представлена на рисунке 2.3.

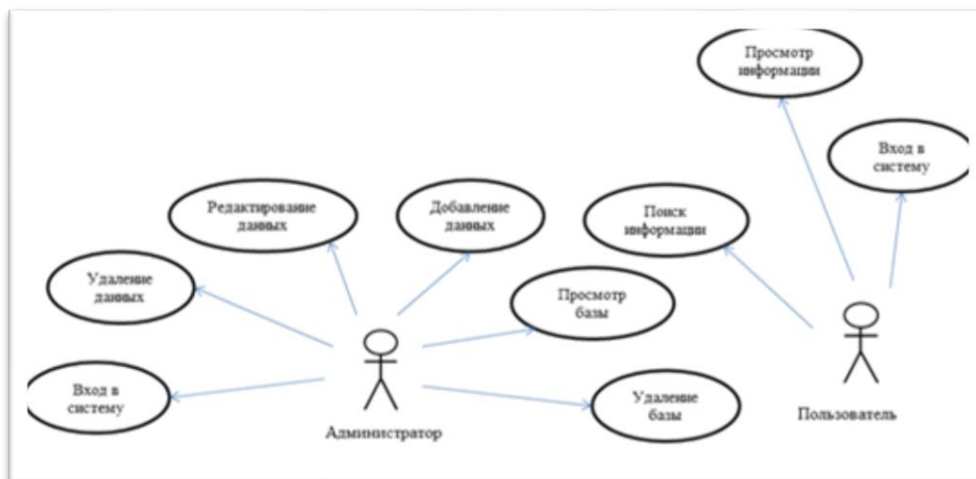


Рисунок 2.3 – Диаграмма прецедентов

Из данной диаграммы видно, что пользоваться базой могут 2 группы пользователей: администратор и пользователь.

2.2.3 Диаграмма классов

Диаграмма классов в UML является частным случаем ER диаграммы. ER диаграммы используются для логического проектирования баз данных. Главное их отличие: В ER диаграмме уделяется внимание структуре данных, а в диаграмме классов – поведению классов. Диаграмма классов представлена на рисунке 2.4.

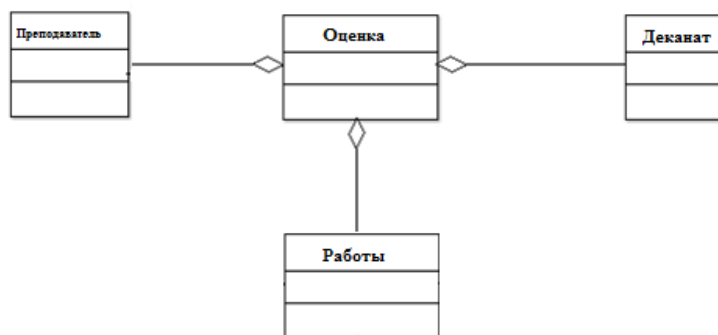


Рисунок 2.4 – Диаграмма классов

2.2.4 Диаграмма развертывания

Диаграмма развертывания показывает принцип работы базы данных и представлена на рисунке 2.5.

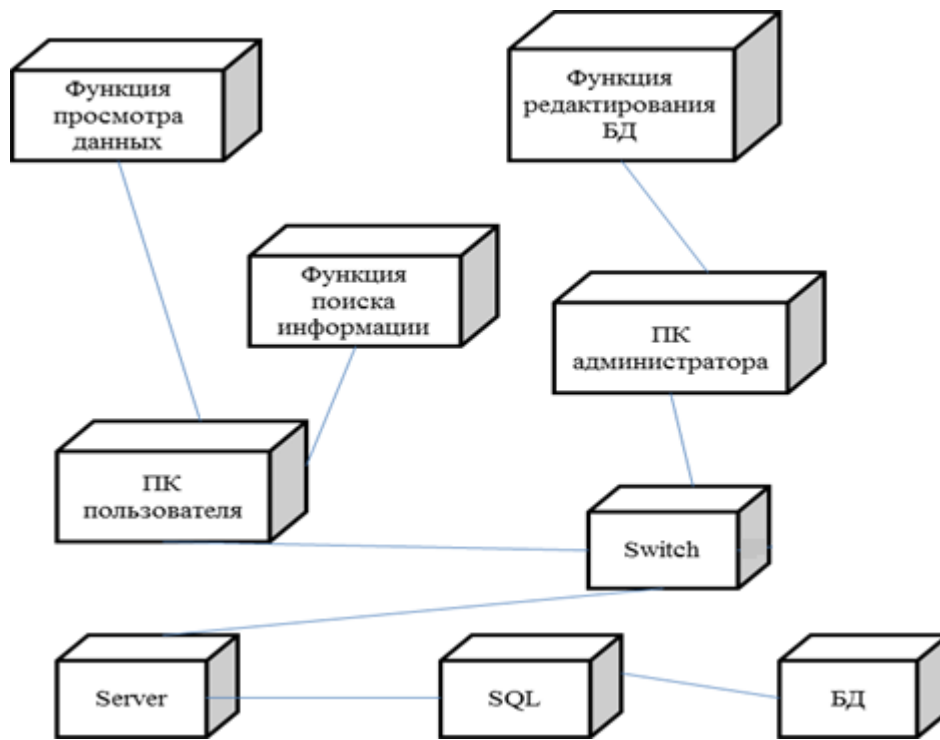


Рисунок 2.5 – Диаграмма развертывания

3 Проектирование БД

3.1 Концептуальное проектирование

Учитывая особенности данной системы, рассмотренной в анализе предметной области, выделим таблицы и связи между ними, представленные на рисунке 3.1.

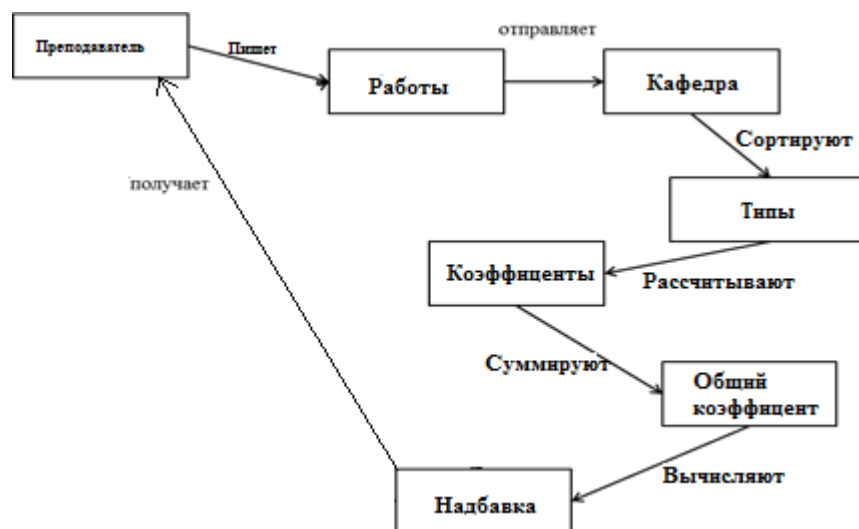


Рисунок 3.1 – Общая ER–диаграмма

На основе этапа начальной разработки сформируем бизнес–правила. Затем определяем связи между сущностями, в основном опираясь на описание операций. Точнее связи сущности базируются на бизнес–правилах, построенных на основе подробного описания операций.

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

Бизнес правило 1

Не может быть двух преподавателей с одинаковыми ID номерами. Преподаватель не может работать на нескольких кафедрах, преподаватель числится только на одной кафедре в данный момент времени. Тем временем на кафедре работает много преподавателей. Не существует кафедры без заведующего кафедрой, поэтому первый вводимый в таблицу преподаватель – зав. кафедры, что реализуется рекурсивной связью. На основе Бизнес правила 1 мы получаем сегмент концептуальной модели, представленной на рисунке 3.2.

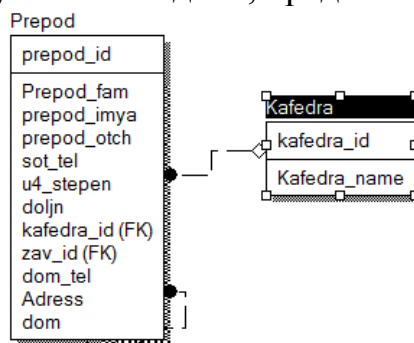


Рисунок 3.2 – ER–диаграмма Бизнес правила 1

Бизнес правило 2

Каждая методическая или научная работа имеет свой вид. Вид определяет количество баллов, начисляемые за эту работу. Одна методическая работа может быть написана несколькими преподавателями с одной или различных кафедр. Для реализации такого вводим дополнительную таблицу «Авторство». На основе Бизнес правила 2 мы получаем следующий сегмент концептуальной модели, представленной на рисунке 3.3.

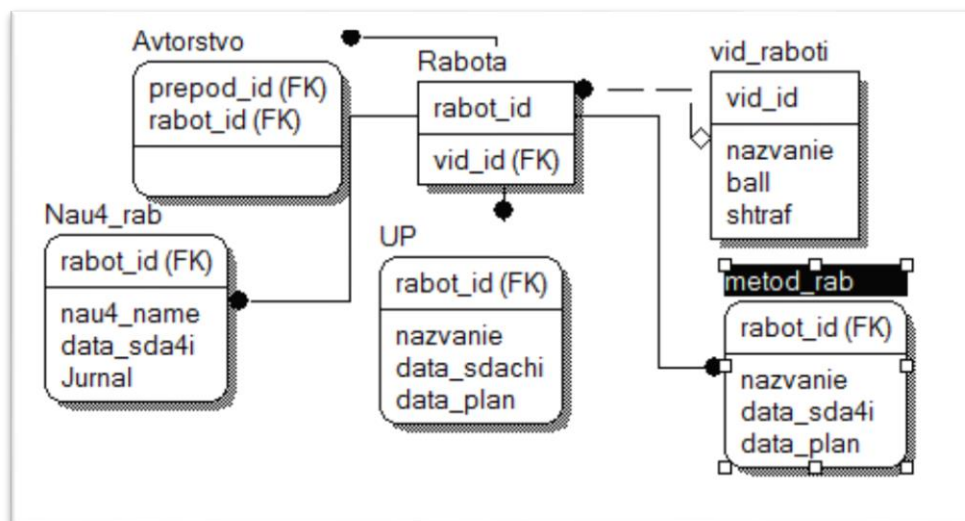


Рисунок 3.3 – ER–диаграмма Бизнес правила 2

Бизнес правило 3

В каждом семестре работа преподавателя оценивается по ряду критериев студентами, преподавателями и заведующим кафедрами. Эти оценки учитываются или принимаются в учет однократно при расчете очередного рейтинга. На основе Бизнес правила 3 мы получаем сегмент концептуальной модели, представленной на рисунке 3.4.

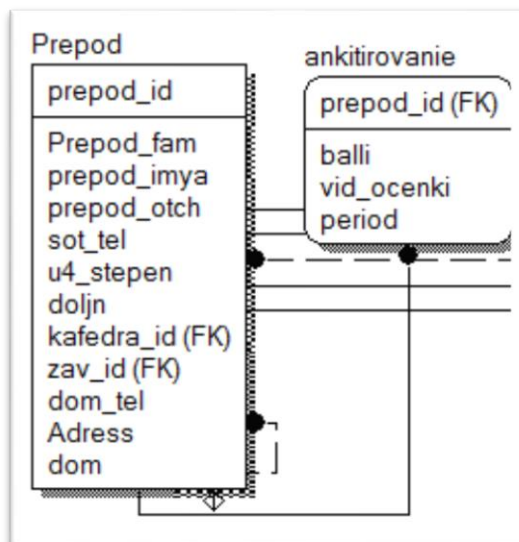


Рисунок 3.4 – ER–диаграмма Бизнес правила 3

Бизнес правило 4

При оценке качества работы преподавателя кафедры физического воспитания учитываются спортивные достижения его студентов. Каждое спортивное достижение имеет определенный вид который определяет количество баллов, начисляемых преподавателю. Каждое спортивное достижение может учитываться при расчете рейтинга только одного преподавателя. Спортивные достижения учитываются в расчете рейтинга только в течение определенного календарного периода. На основе Бизнес правила 4 мы получаем сегмент концептуальной модели, представленной на рисунке 3.5.

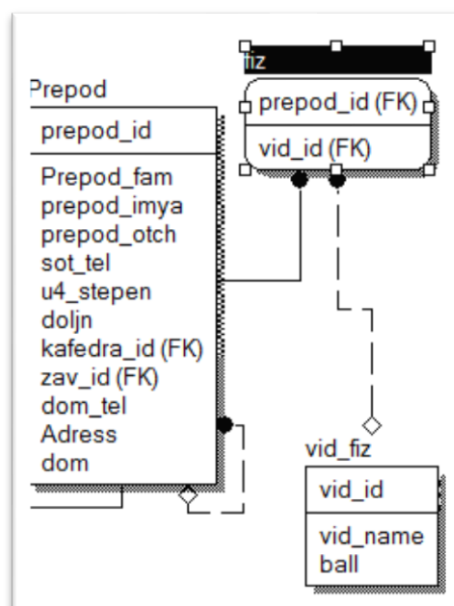


Рисунок 3.5 – ER–диаграмма Бизнес правила 4

Бизнес правило 5

Для каждого преподавателя определяется набор коэффициентов участвующих в расчете рейтинга ее преподавателей. На основе Бизнес правила 5 мы получаем сегмент концептуальной модели, представленной на рисунке 3.6.

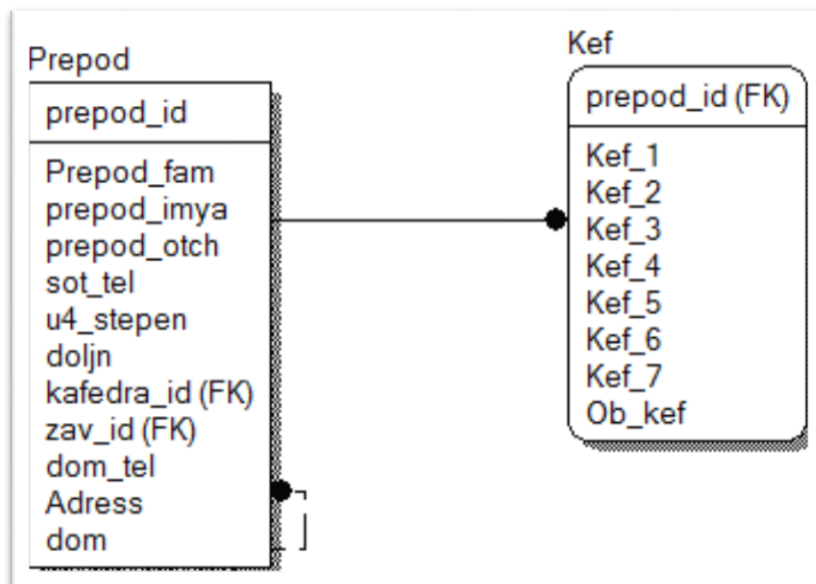


Рисунок 3.6 – ER–диаграмма Бизнес правила 5

На рисунке 3.7 представлен окончательный вариант концептуальной модели информационной системы библиотеки.

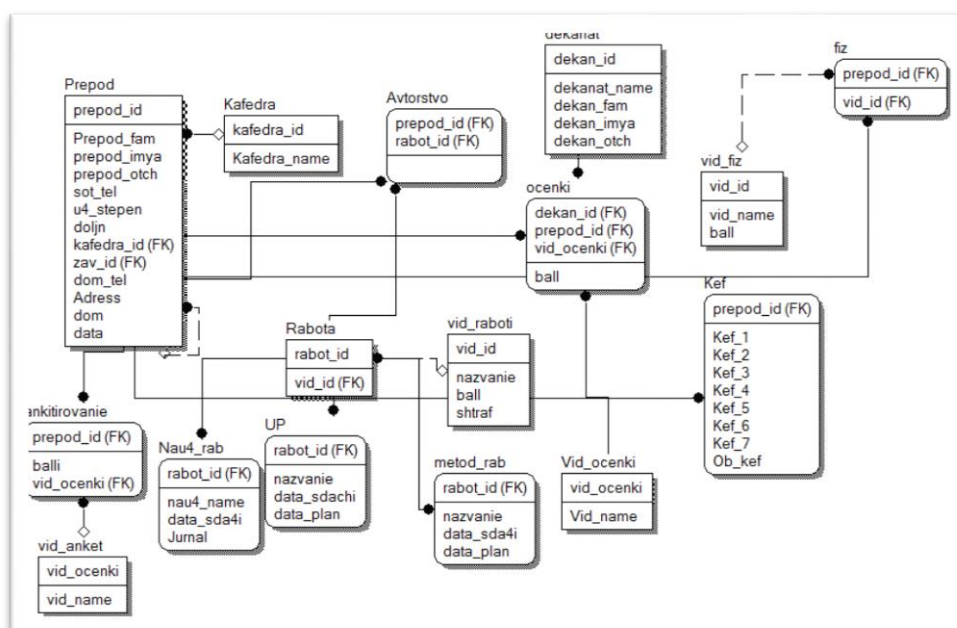


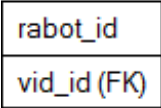
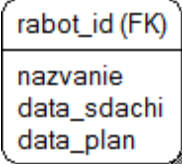
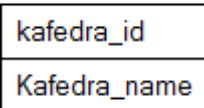
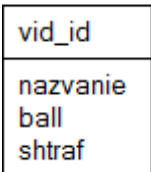
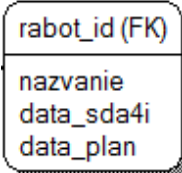
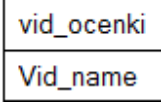
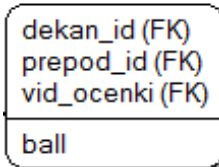

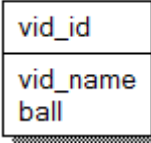
Рисунок 3.7 – Окончательный вариант концептуальной модели информационной системы библиотеки

3.2 Определение атрибутов каждой сущности

На таблице 3.1 показано описание каждой таблицы базы данных.

Т а б л и ц а 3.1 – Описание таблиц

| Сегмент ER модели | Описание |
|---|---|
| <p>Prepod</p> <pre> prepod_id Prepod_fam prepod_imya prepod_otch sot_tel u4_stepen doljn kafedra_id (FK) zav_id (FK) dom_tel Adress dom data </pre> | <p>Таблица Преподаватели содержит следующие данные:</p> <p>Prepod_ID – уникальный код преподавателя. Prepod_fam – фамилия преподавателя. Prepod_imya – имя преподавателя. Prepod_otch – отчество преподавателя. Sot_tel – сотовый телефон преподавателя. u4_stepen – ученая степень преподавателя. doljn – должность преподавателя. kafedra_id – уникальный номер кафедры. zav_id – уникальный номер заведующего кафедрой. dom_tel – домашний телефон преподавателя. adress – Адрес преподавателя. dom – дом преподавателя. data – дата рождения преподавателя.</p> |
| <p>ankitirovanie</p> <pre> prepod_id (FK) balli vid_ocenki period </pre> | <p>Таблица Анкетирование содержит следующие данные:</p> <p>Prepod_id – уникальный код преподавателя. balli – баллы. vid_ocenki – уникальный код вида оценок за анкетирование. period – период, за который были выставлены баллы.</p> |
| <p>vid_anket</p> <pre> vid_ocenki vid_name </pre> | <p>Таблица Виды анкетирование содержит следующие данные:</p> <p>vid_ocenki – Уникальный код вида оценок за анкетирование. vid_name – название вида оценок за анкетирование.</p> |
| <p>Nau4_rab</p> <pre> rabot_id (FK) nau4_name data_sda4i Jurnal </pre> | <p>Таблица Научная работа содержит следующие данные:</p> <p>rabot_ID – уникальный код работы. nau4_NAME – название работы. data_sda4i – дата сдачи работы. jurnal – Журнал, в котором была опубликована работа</p> |
| <p>Avtorstvo</p> <pre> prepod_id (FK) rabot_id (FK) </pre> | <p>Таблица Авторство содержит следующие данные:</p> <p>prepod_id – уникальный код преподавателя. rabot_id – уникальный код работы.</p> |

| Сегмент ER модели | Описание |
|---|--|
| <p>Rabota</p>  | <p>Таблица Работа содержит следующие данные: rabot_id – уникальный код работы. vid_id – уникальный код вида работы.</p> |
| <p>UP</p>  | <p>Таблица Учебное пособие содержит следующие данные: rabot_id – уникальный код работы. nazvanie – название работы. data_sdachi – дата сдачи работы. data_plan – плановая дата сдачи работы.</p> |
| <p>Kafedra</p>  | <p>Таблица Кафедра содержит следующие данные: kafedra_ID – уникальный код кафедры. kafedra_name – название кафедры.</p> |
| <p>vid_raboti</p>  | <p>Таблица Виды работ содержит следующие данные: vid_ID – уникальный код вида работы. nazvanie – название вида работы. ball – баллы за работу. shtraf – штраф за работу.</p> |
| <p>metod_rab</p>  | <p>Таблица Методические работы программы содержит следующие данные: rabot_id – уникальный код работы. nazvanie – название работы. data_sdachi – дата сдачи работы. data_plan – плановая дата сдачи работы.</p> |
| <p>Vid_ocenki</p>  | <p>Таблица Виды оценок содержит следующие данные: vid_ocenki – уникальный код вида оценки. vid_NAME – название вида оценки.</p> |
| <p>ocenki</p>  | <p>Таблица Оценка содержит следующие данные: dekan_ID – уникальный код деканата. Prepod_ID – уникальный код преподавателя. vid_ocenki – уникальный код вида оценки. ball – балл, выставленным деканатом.</p> |
| <p>fiz</p>  | <p>Таблица Оценка за физ. воспитание содержит следующие данные: Prepod_ID – уникальный код преподавателя. vid_id – идентификационный номер</p> |
| <p>vid_fiz</p>  | <p>Таблица Виды оценок за физ. воспитание содержит следующие данные: vid_ID – уникальный код вида физ. работ. vid_name – название вида физ. Работ ball – балл за физ. работу</p> |

| Сегмент ER модели | Описание |
|---|---|
| <div style="border: 1px solid black; padding: 5px; width: fit-content;"> <div style="border: 1px solid black; padding: 2px; margin-bottom: 5px;">dekanat</div> <div style="border: 1px solid black; padding: 2px; margin-bottom: 2px;">dekan_id</div> <div style="border: 1px solid black; padding: 2px;">dekanat_name dekan_fam dekan_imya dekan_otch</div> </div> | <p>Таблица Деканат содержит следующие данные:</p> <p>dekan_ID – уникальный код деканата.</p> <p>dekanat_name – название деканата.</p> <p>dekan_fam – фамилия декана.</p> <p>dekan_imya – имя декана.</p> <p>dekan_otch – отчество декана.</p> |

3.3 Составление реляционных отношений

Каждое реляционное отношение соответствует одной сущности и в него вносятся все атрибуты сущности. Для каждого отношения необходимо определить первичный ключ и внешние ключи (если они есть).

Отношения приведены в таблицах 3.2–3.17:

Т а б л и ц а 3.2 – Схема отношения преподаватель (prepod)

| Содержание поля | Имя поля | Тип, длина | Примечание |
|--|-------------|--------------|--------------------------------|
| уникальный код преподавателя | Prepod_ID | bigint | Первичный ключ |
| фамилия преподавателя | Prepod_fam | varchar (18) | not null |
| имя преподавателя | Prepod_imya | varchar (18) | not null |
| отчество преподавателя. | Prepod_otch | varchar (18) | null |
| сотовый телефон преподавателя. | Prepod_tel | bigint | null |
| ученая степень преподавателя. | u4_stepen | varchar (18) | null |
| должность преподавателя. | doljnost | varchar (18) | not null |
| уникальный номер кафедры. | kafedra_id | int | Внешний ключ (к Кафедре) |
| уникальный номер заведующего кафедрой. | zav_id | bigint | Внешний ключ (к Преподавателю) |
| домашний телефон преподавателя. | dom_tel | bigint | null |
| Адрес преподавателя. | adress | varchar (30) | null |
| дом преподавателя. | dom | varchar (10) | null |
| дата рождения преподавателя. | data | datetime | null |

Т а б л и ц а 3.3 – Схема отношения Виды Работ (Vid)

| Содержание поля | Имя поля | Тип, длина | Примечание |
|-----------------------------|----------|---------------|----------------|
| уникальный код вида работы. | vid_ID | int | Первичный ключ |
| название вида работы. | vid_name | varchar (150) | not null |
| баллы за работу. | ball | int | null |
| штраф за работу. | shtraf | int | null |

Т а б л и ц а 3.4 – Схема отношения Методическая работа (metod)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|-----------------------------|-----------------|-------------------|---|
| уникальный код работы. | rabot_id | int | Первичный ключ Внешний ключ (к Работе) |
| название работы. | nazvanie | varchar (200) | null |
| дата сдачи работы. | data_sdachi | datetime | not null |
| плановая дата сдачи работы. | data_plan | datetime | null |

Т а б л и ц а 3.5 – Схема отношения Научная работа (Nauchrab)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|------------------------|-----------------|-------------------|-------------------|
| уникальный код работы. | rabot_ID | int | Первичный ключ |
| название работы. | nau4_NAME | varchar(18) | null |
| дата сдачи работы. | data_sda4i | datetime | not null |

Т а б л и ц а 3.6 – Схема отношения Учебное пособие (UP)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|-----------------------------|-----------------|-------------------|-------------------------|
| уникальный код работы. | rabot_id | int | Внешний ключ (к Работе) |
| название работы. | nazvanie | varchar (18) | not null |
| дата сдачи работы. | data_sdachi | datetime | not null |
| плановая дата сдачи работы. | data_plan | datetime | null |

Т а б л и ц а 3.7 – Схема отношения Анкетирование (Anketirovanie)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|--|-----------------|-------------------|-------------------------------------|
| уникальный код преподавателя. | Prepod_id | bigint | Внешний ключ (к Преподавателю) |
| баллы. | balli | float | not null |
| Уникальный код вида оценок за анкетирование. | vid_ocenki | int | Внешний ключ (к Виду анкетирования) |

Т а б л и ц а 3.8 – Схема отношения Виды анкетирование(vid_anket)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|--|-----------------|-------------------|-------------------|
| Уникальный код вида оценок за анкетирование. | vid_ocenki | int | Первичный ключ |
| название вида оценок за анкетирование | vid_name | varchar (90) | not null |

Т а б л и ц а 3.9 – Схема отношения Авторство (Avtorstvo)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|------------------------|-----------------|-------------------|--------------------------------|
| Код преподавателя. | prepod_id | bigint | Внешний ключ (к Преподавателю) |
| Код работы. | rabot_id | int | Внешний ключ (к Работе) |

Т а б л и ц а 3.10 – Схема отношения Работа (Rabota)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|-----------------------------|-----------------|-------------------|------------------------------|
| уникальный код работы. | rabot_id | int | Первичный ключ |
| уникальный код вида работы. | vid_id | int | Внешний ключ (к Видам работ) |

Т а б л и ц а 3.11 – Схема отношения Кафедра (Kafedra)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|-------------------------|-----------------|-------------------|-------------------|
| уникальный код кафедры. | kafedra_ID | int | Первичный ключ |
| название кафедры. | kafedra_name | varchar (18) | not null |

Т а б л и ц а 3.12 – Схема отношения Виды оценок (Vid_ocenok)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|-----------------------------|-----------------|-------------------|-------------------|
| уникальный код вида оценки. | vid_ocenki | int | Первичный ключ |
| название вида оценки. | vid_NAME | varchar(90) | not null |

Т а б л и ц а 3.13 – Схема отношения Оценка (Ocenka)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|-------------------------------|-----------------|-------------------|--|
| уникальный код деканата. | dekan_ID | int | Первичный ключ Внешний ключ (к Деканату) |
| уникальный код преподавателя. | Prepod_ID | bigint | Первичный ключ Внешний ключ (к Преподавателю) |
| уникальный код вида оценки. | vid_ocenki | int | Первичный ключ Внешний ключ (к Виду оценок) |
| балл, выставленным деканатом. | ball | int | not null |

Т а б л и ц а 3.14 – Схема отношения Оценка за физ. воспитание(fiz)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|---|-----------------|-------------------|--|
| уникальный код преподавателя. | Prepod_ID | bigint | Первичный ключ Внешний ключ (к Преподавателю) |
| идентификационный номер вида физ. работ | vid_id | int | Внешний ключ (к Виду физ. воспитания) |

Т а б л и ц а 3.15 – Схема отношения Виды физ. воспитания (vid_fiz)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|---------------------------------|-----------------|-------------------|-------------------|
| уникальный код вида физ. работ. | vid_ID | int | Первичный ключ |
| название вида физ. Работ | vid_name | varchar (100) | not null |
| балл за физ. работу | ball | int | null |

Т а б л и ц а 3.16 – Схема отношения Деканат (Dekanat)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|--------------------------|-----------------|-------------------|-------------------|
| уникальный код деканата. | dekan_ID | int | Первичный ключ |
| название деканата. | dekanat_name | varchar (18) | null |
| фамилия декана. | dekan_fam | varchar (18) | not null |
| имя декана. | dekan_imya | varchar (18) | not null |
| отчество декана. | dekan_otch | varchar (18) | null |

Т а б л и ц а 3.17 – Схема отношения Коэффициенты (Kef)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
|-------------------------------|-----------------|-------------------|---|
| уникальный код преподавателя. | Prepod_ID | bigint | Первичный ключ Внешний ключ (к Преподавателю) |
| первый коэффициент. | Kef_1 | float | null |
| второй коэффициент. | Kef_2 | float | null |
| третий коэффициент. | Kef_3 | float | null |
| четвертый коэффициент. | Kef_4 | float | null |
| пятый коэффициент. | Kef_5 | float | null |
| шестой коэффициент. | Kef_6 | float | null |
| седьмой коэффициент. | Kef_7 | float | null |
| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Примечание</i> |
| общий коэффициент. | Ob_kef | float | null |

3.4 Расчет места для хранения БД

На этом этапе, необходимо знать какой объем памяти будет занимать создаваемая база данных. Объем внешней памяти, необходимый для функционирования системы, складывается из двух составляющих: память, занимаемая модулями СУБД (ядро, утилиты, вспомогательные программы), и память, отводимая под данные (Мд). Наиболее существенным обычно является Мд.

Объем памяти, занимаемый программными модулями пользователя, обычно невелик по сравнению с объемом самих данных, поэтому может не учитываться. В проекте рассчитывается предполагаемый максимальный объем памяти занимаемой БД. Расчет места для хранения БД описаны в таблицах 3.18–3.33.

Т а б л и ц а 3.18 – Расчет физической памяти для таблицы преподаватель (prepod)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|--|-----------------|-------------------|--------------------|
| уникальный код преподавателя | Prepod_ID | bigint | 8 |
| фамилия преподавателя | Prepod_fam | varchar (18) | 18 |
| имя преподавателя | Prepod_imya | varchar (18) | 18 |
| отчество преподавателя. | Prepod_otch | varchar (18) | 18 |
| сотовый телефон преподавателя. | Prepod_tel | bigint | 8 |
| ученая степень преподавателя. | u4_stepen | varchar (18) | 18 |
| должность преподавателя. | doljnost | varchar (18) | 18 |
| уникальный номер кафедры. | kafedra_id | int | 4 |
| уникальный номер заведующего кафедрой. | zav_id | bigint | 8 |
| домашний телефон преподавателя. | dom_tel | bigint | 8 |

Общая длина строки: 38 байт
 Число строк: ~ 38
 Общий объем требуемой памяти: ~ 1444 байт

Т а б л и ц а 3.19 – Расчет физической памяти для таблицы Анкетирование (Anketirovanie)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|--|-----------------|-------------------|--------------------|
| уникальный код преподавателя. | Prepod_id | bigint | 8 |
| баллы. | balli | float | 8 |
| Уникальный код вида оценок за анкетирование. | vid_ocenki | int | 4 |

Общая длина строки: 20 байт
 Число строк: ~ 76
 Общий объем требуемой памяти: ~ 1520 байт

Т а б л и ц а 3.20 – Расчет физической памяти для таблицы Виды анкетирование(vid_anket)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|--|-----------------|-------------------|--------------------|
| Уникальный код вида оценок за анкетирование. | vid_ocenki | int | 4 |
| название вида оценок за анкетирование | vid_name | varchar (90) | 90 |

Общая длина строки: 94 байт
 Число строк: ~ 2
 Общий объем требуемой памяти: ~ 188 байт

Т а б л и ц а 3.21 – Расчет физической памяти для таблицы Научная работа

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|------------------------|-----------------|-------------------|--------------------|
| уникальный код работы. | rabot_ID | int | 4 |
| название работы. | nau4_NAME | varchar(18) | 18 |
| дата сдачи работы. | data_sda4i | datetime | 8 |

Общая длина строки: 30 байт
 Число строк: ~ 50
 Общий объем требуемой памяти: ~ 1500 байт

Т а б л и ц а 3.22 – Расчет физической памяти для таблицы Учебное пособие

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|-----------------------------|-----------------|-------------------|--------------------|
| уникальный код работы. | rabot_id | int | 4 |
| название работы. | nazvanie | varchar (18) | 18 |
| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
| дата сдачи работы. | data_sdachi | datetime | 8 |
| плановая дата сдачи работы. | data_plan | datetime | 8 |

Общая длина строки: 38 байт

Число строк: ~ 10

Общий объем требуемой памяти: ~ 380 байт

Т а б л и ц а 3.23 – Расчет физической памяти для таблицы Авторство (Avtorstvo)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|-------------------------------|-----------------|-------------------|--------------------|
| уникальный код преподавателя. | prepod_id | bigint | 8 |
| уникальный код работы. | rabot_id | int | 4 |

Общая длина строки: 12 байт

Число строк: ~ 100

Общий объем требуемой памяти: ~ 1200 байт

Т а б л и ц а 3.24 – Расчет физической памяти для таблицы Работа (Rabota)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|-----------------------------|-----------------|-------------------|--------------------|
| уникальный код работы. | rabot_id | int | 4 |
| уникальный код вида работы. | vid_id | int | 4 |

Общая длина строки: 8 байт

Число строк: ~ 60

Общий объем требуемой памяти: ~ 480 байт

Т а б л и ц а 3.25 – Расчет физической памяти для таблицы Кафедра (Kafedra)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|-------------------------|-----------------|-------------------|--------------------|
| уникальный код кафедры. | kafedra_ID | int | 4 |
| название кафедры. | kafedra_name | varchar (18) | 18 |

Общая длина строки: 22 байт

Число строк: ~ 10

Общий объем требуемой памяти: ~ 220 байт

Т а б л и ц а 3.26 – Расчет физической памяти для таблицы Виды Работ (Vid)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|-----------------------------|-----------------|-------------------|--------------------|
| уникальный код вида работы. | vid_ID | int | 4 |
| название вида работы. | vid_name | varchar (150) | 150 |
| баллы за работу. | ball | int | 4 |
| штраф за работу. | shtraf | int | 4 |

Общая длина строки: 162 байт

Число строк: ~ 47

Общий объем требуемой памяти: ~ 7614 байт

Т а б л и ц а 3.27 – Расчет физической памяти для таблицы Методическая работа (metod)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина (байт)</i> |
|-----------------------------|-----------------|-------------------|---------------------|
| уникальный код работы. | rabot_id | int | 4 |
| название работы. | nazvanie | varchar (200) | 200 |
| дата сдачи работы. | data_sdachi | datetime | 8 |
| плановая дата сдачи работы. | data_plan | datetime | 8 |

Общая длина строки: 220 байт

Число строк: ~ 40

Общий объем требуемой памяти: ~ 8800 байт

Т а б л и ц а 3.28 – Расчет физической памяти для таблицы Оценка (Ocenka)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|-------------------------------|-----------------|-------------------|--------------------|
| уникальный код деканата. | dekan_ID | int | 4 |
| уникальный код преподавателя. | Prepod_ID | bigint | 8 |
| уникальный код вида оценки. | vid_ocenki | int | 4 |
| балл, выставленным деканатом. | ball | int | 4 |

Общая длина строки: 20 байт

Число строк: ~ 38

Общий объем требуемой памяти: ~ 760 байт

Т а б л и ц а 3.29 – Расчет физической памяти для таблицы Виды оценок (Vid_ocenok)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|-----------------------------|-----------------|-------------------|--------------------|
| уникальный код вида оценки. | vid_ocenki | int | 4 |
| название вида оценки. | vid_NAME | varchar(90) | 90 |

Общая длина строки: 94 байт

Число строк: ~ 2

Общий объем требуемой памяти: ~ 188 байт

Т а б л и ц а 3.30 – Расчет физической памяти для таблицы Оценка за физ. воспитание(fiz)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|---|-----------------|-------------------|--------------------|
| уникальный код преподавателя. | Prepod_ID | bigint | 8 |
| идентификационный номер вида физ. работ | vid_id | int | 4 |

Общая длина строки: 12 байт

Число строк: ~ 10

Общий объем требуемой памяти: ~ 120 байт

Т а б л и ц а 3.31 – Расчет физической памяти для таблицы Виды физ. воспитания (vid_fiz)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|---------------------------------|-----------------|-------------------|--------------------|
| уникальный код вида физ. работ. | vid_ID | int | 4 |
| название вида физ. Работ | vid_name | varchar (100) | 100 |
| балл за физ. работу | ball | int | 4 |

Общая длина строки: 108 байт

Число строк: ~ 16

Общий объем требуемой памяти: ~ 1728 байт

Т а б л и ц а 3.32 – Расчет физической памяти для таблицы Деканат (Dekanat)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|--------------------------|-----------------|-------------------|--------------------|
| уникальный код деканата. | dekan_ID | int | 4 |
| название деканата. | dekanat_name | varchar (18) | 18 |
| фамилия декана. | dekan_fam | varchar (18) | 18 |
| имя декана. | dekan_imya | varchar (18) | 18 |
| отчество декана. | dekan_otch | varchar (18) | 18 |

Общая длина строки: 76 байт

Число строк: ~ 10

Общий объем требуемой памяти: ~ 760 байт

Т а б л и ц а 3.33 – Расчет физической памяти для таблицы Коэффициенты (Kef)

| <i>Содержание поля</i> | <i>Имя поля</i> | <i>Тип, длина</i> | <i>Длина(байт)</i> |
|-------------------------------|-----------------|-------------------|--------------------|
| уникальный код преподавателя. | Prepod_ID | bigint | 8 |
| первый коэффициент. | Kef_1 | float | 8 |
| второй коэффициент. | Kef_2 | float | 8 |
| третий коэффициент. | Kef_3 | float | 8 |
| четвертый коэффициент. | Kef_4 | float | 8 |
| пятый коэффициент. | Kef_5 | float | 8 |
| шестой коэффициент. | Kef_6 | float | 8 |
| седьмой коэффициент. | Kef_7 | float | 8 |
| общий коэффициент. | Ob_kef | float | 8 |

Общая длина строки: 72 байт

Число строк: ~ 38

Общий объем требуемой памяти: ~ 2736 байт

Таким образом, из полученных данным можно рассчитать приблизительный максимальный объем базы данных. Объем будет равен сумме объемов всех таблиц:

База данных = Судья + Тренер + Хореограф + Заявка Фигурист + Заявка Судьи + Категория + Медаль + Элемент + Программа + Фигурист + Оценка за технику + О соревновании + О программе + Пол + Оценка за вид программы + Итоговая оценка = ***1444 + 1520 + 188 + 1500 + 1200 + 480 + 380 + 220 + 7614 + 8800 + 188 + 760 + 120 + 1728 + 760 + 2736 = 29638 (байт)***

Объём памяти, занимаемый программными модулями пользователя, обычно невелик по сравнению с объёмом самих данных, поэтому может не учитываться. Требуемый объём оперативной памяти определяется на основании анализа интенсивности запросов и объёма результирующих данных.

4 Логическое проектирование

4.1 Схемы отношений составленные на языке определения данных (ddl, datadefinitionlanguage)

4.1.1 Создание таблиц

```
--Преподаватель
createtableprepodschem.prepod(
  prepod_id      bigintprimarykey,
  prepod_fam     varchar(18)notnull,
  prepod_imya    varchar(18)notnull,
  prepod_otch    varchar(18),
  prepod_tel     bigint,
  u_stepen       varchar(18),
  doljnost       varchar(18)notnull,
  kafedra_id     intreferencesprepodschem.kafedra(kafedra_id),
  zav_id         bigintnotnull,
  constraintfk_prepod_prepodforeignkey(zav_id)referencesprepodschem.prepod(
  prepod_id));
--Кафедра
createtableprepodschem.kafedra(
  kafedra_id     intprimarykey,
  kafedra_name   varchar(18));
--Анкетирование
createtableocenkaschem.anketirovanie(
  prepod_id      bigintnotnull,
  balli          intnotnull,
  vid_ocenki     intnotnullreferencesvid_anket(vid_ocenki),
  constraintfk_anketirovanie_prepodforeignkey(prepod_id)referencesprepodsche
ma.prepod(prepod_id),
  constraintpk_anketirovanieprimarykey (prepod_id,vid_ocenki));
--ВидыАнкетирования
createtablevid_anket(
  vid_ocenki    int      primarykey,
  vid_name      varchar(20)notnull);
--Научнаяработа
createtableprepodschem.nauchrab(
  rabota_id     intnotnull,
  nauch_name    varchar(18),
  data_sdachi   datetimenotnull,
  jurnal        varchar(30),
  constraintfk_nauchrab_rabotaforeignkey(rabota_id)referencesrabota(rabota_i
d),
  constraintpk_nauchrabprimarykey (rabota_id)
);
--ВидыФиз.воспитания
createtablevid_fiz(
  vid_id        intprimarykey,
  vid_name      varchar(90)notnull,
  ball          intnotnull);
--Видыработ
createtablevid(
  vid_id        intprimarykey,
  vid_name      varchar(18)notnull,
  ball          int,
  shtraf        int);
--Методическая работа
```

```

createtableprepodschema.metod(
  rabota_id      intnotnullreferencesrabota(rabota_id),
  nazvanie      varchar(18)notnull,
  data_sdachi   datetime,
  data_plan     datetimenotnull,
  constraintpk_metodprimarykey(rabota_id)
);
--Работа
createtablerabota(
  rabota_id     intprimarykey,
  vid_id        intnotnullreferencesvid(vid_id))
--Авторство
createtableavtorstvo(
  prepod_id     bigintnotnullreferencesprepodschema.prepod(prepod_id),
  rabota_id     intnotnullreferencesrabota(rabota_id),
  constraintpk_avtorstvoprimarykey(prepod_id,rabota_id));
droptableavtorstvo
--Физ. воспитание
createtablefiz(
  prepod_id     bigintnotnull,
  vid_id        intnotnullreferencesvid_fiz(vid_id),
  constraintfk_fiz_prepodforeignkey(prepod_id)referencesprepodschema.prepod(
prepod_id),
  constraintpk_fizprimarykey(prepod_id,vid_id));
--Учебноепособие
createtableUP(
  rabota_id     intnotnullreferencesrabota(rabota_id),
  nazvanie      varchar(18)notnull,
  data_sdachi   datetimenotnull,
  data_plan     datetime
  constraintpk_upprimarykey (rabota_id))

createtabledekanat(
  dekan_id      bigintprimarykey,
  dekanat_name  varchar(18),
  dekan_fam     varchar(18)notnull,
  dekan_imya    varchar(18)notnull,
  dekan_otch    varchar(18))
--Оценкидеканата
createtableocenki(
  dekan_id      bigintnotnullreferencesdekanat(dekan_id),
  prepod_id     bigintnotnullreferencesprepodschema.prepod(prepod_id),
  ball          int notnull,
  vid_id        intnotnullreferencesvid_ocenki(vid_id),
  constraintpk_ocenkiprimarykey(dekan_id,prepod_id,vid_id))
--Видыоценокдеканата
createtablevid_ocenki(
  vid_id        intprimarykey,
  vid_name      varchar(50)notnull)
--Коэффициенты
createtablekef(
  prepod_id     bigintnotnullreferencesprepodschema.prepod(prepod_id),
  kef_1         intnotnull,
  kef_2         intnotnull,
  kef_3         intnotnull,
  kef_4         intnotnull,
  kef_5         intnotnull,
  kef_6         intnotnull,
  kef_7         intnotnull,
  Ob_kef        intnotnull,
  constraintpk_kefprimarykey (prepod_id))

```

В результате выполнения созданных скриптов, получим реальную БД. Диаграмма физической модели данной базы данных представлена на рисунке 4.1.

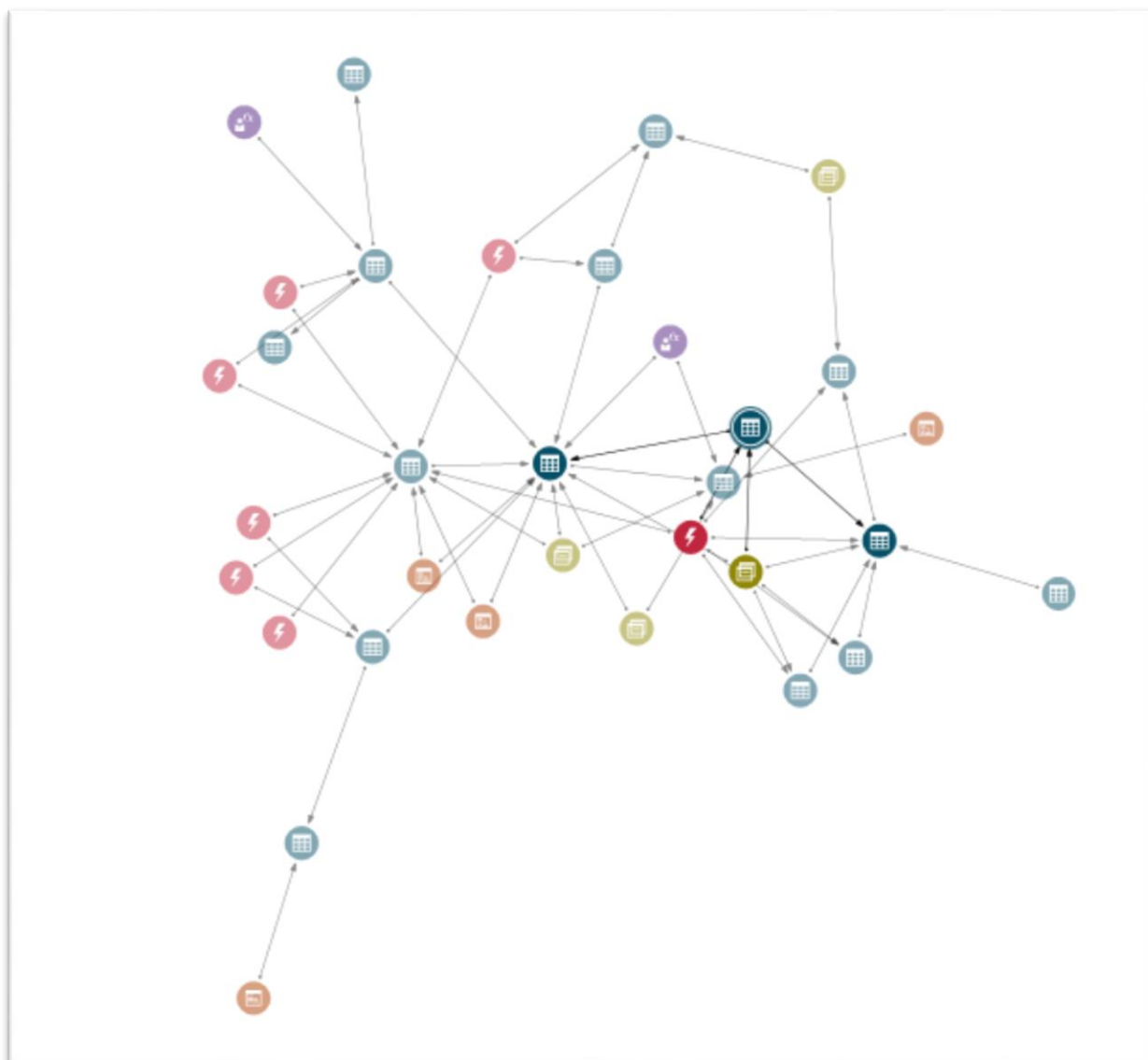


Рисунок 4.1 – Диаграмма физической модели данной базы данных

4.2 Заполнение созданной БД

Созданная БД содержит все еще пустые табличные структуры. Для заполнения базы данных был использован SQL скрипт написанный вручную.

4.2.1 Заполнение таблицы Виды анкетирования

```
INSERT INTO vid_anket
VALUES
(1, 'Анкетирование студентами'),
```

(2, 'Анкетирование преподавателями');

4.2.2 Заполнение таблицы Авторство

```
INSERTINTOAvtorstvo
VALUES
(620002,4001), (620002,4002), (620020,4010),
(620021,4011), (620021,4012), (620021,4013),
(620022,4014), (620022,4015), (620022,4016),
(620022,4017), (620022,4018), (620004,4019),
(620004,4020), (620004,4021), (620004,4022),
(620004,4023), (620004,4024), (620004,4025),
(620004,4026), (620006,4003), (620006,4004),
(620007,4002), (620007,4005), (620007,4006),
(620008,4007), (620010,4008), (620012,4009),
(620012,4031), (620013,4031), (620018,4032),
(620018,4033), (620018,4034), (620018,4035),
(620018,4036), (620018,4037), (620018,4038),
(620018,4039), (620018,4040), (620018,4141),
(620018,4242), (620018,4343), (620018,4444),
(620018,4445), (620020,4027), (620020,4028),
(620020,4029), (620020,4030), (620021,4042),
(620021,4043), (620021,4044), (620022,4032),
(620022,4033), (620022,4034), (620022,4035),
(620022,4036), (620022,4037), (620022,4039),
(620022,4041), (620022,4141), (620022,4242),
(620022,4343), (620022,4444), (620022,4445),
(620023,4032), (620023,4033), (620023,4034),
(620023,4035), (620023,4036), (620023,4037),
(620023,4040), (620023,4041), (620023,4141),
(620023,4242), (620023,4343), (620023,4444),
(620023,4445), (620025,4039), (620025,4040),
(620025,4041), (620032,4045), (620032,4046),
(620032,4047), (620032,4048);
```

4.2.3 Заполнение таблицы Деканат

```
INSERTINTODekanat
VALUES
(1, 'ФИТ', 'Табултаев', 'Саин', 'Сейдинкожаевич');
```

4.2.4 Заполнение таблицы Физ. воспитания

```
INSERTINTOfiz
VALUES
(620001,2049),
(620015,2062);
```

4.2.5 Заполнение таблицы Оценки деканата

```
INSERTINTOocenki
VALUES
(453201,620004,2,2),
(453201,620025,2,2),
(453201,620029,2,2);
```

4.2.6 Заполнение таблицы Работа

```
INSERTINTOrabota
VALUES
(4001,2009), (4002,2007), (4003,2003),
(4004,2002), (4005,2008), (4006,2008),
(4007,2002), (4008,2009), (4009,2007),
(4010,2008), (4011,2008), (4012,2008),
(4013,2008), (4014,2008), (4015,2008),
(4016,2008), (4017,2008), (4018,2008),
(4019,2008), (4020,2008), (4021,2002),
(4022,2002), (4023,2002), (4024,2009),
(4025,2010), (4026,2009), (4027,2008),
(4028,2008), (4029,2008), (4030,2008),
(4031,2008), (4032,2008), (4033,2008),
(4034,2008), (4035,2008), (4036,2008),
(4037,2008), (4038,2008), (4039,2008),
(4040,2008), (4041,2008), (4042,2008),
(4043,2008), (4044,2009), (4045,2008),
(4046,2008), (4047,2008), (4048,2008),
(4141,2008), (4242,2003), (4343,2003),
(4444,2003), (4445,2003);
```

4.2.7 Заполнение таблицы Виды работ

```
insertintovid
values (2001, 'Учебник', 80, null),
(2002, 'Учебное пособие', 60, null),
(2003, 'Конспект лекций', 40, null),
(2004, 'Участие в Разработке ГОСО', 40, null),
(2005, 'Участие в разработке типовых программ', 40, null),
(2006, 'Участие в разработке дисциплин', 40, null),
(2007, 'Участие в разработке РУП', 40, null),
(2008, 'Методические Указания', 25, 12),
(2009, 'Постановка новой лаб. работы(физ. установка)', 80, null),
(2010, 'Постановка новой лаб. работ(вирт. работа)', 40, null),
(2011, 'Руководитель темы бюджетных программ', 60, null),
(2012, 'Руководитель темы хоздоговорные НИР и ОКР', 35, null),
(2013, 'Руководитель темы не финансируемые обязательные НИР, ППС', 25, null),
(2014, 'Исполнительные темы бюджетных программ', 30, null),
(2015, 'Исполнительные темы хоздоговорные НИР и ОКР', 20, null),
(2016, 'Исполнительные темы не финансируемые обязательные НИР,
ППС', 15, null),
(2017, 'Защита докторской диссертации', 75, null),
(2018, 'Защита докторской PhD диссертации', 60, null),
(2019, 'Защита кандидатской диссертации', 50, null),
(2020, 'Научное руководство кандидатской диссертации', 50, null),
(2021, 'Научное консультирование по докторской диссертации', 50, null),
(2022, 'Опубликованная научная монография в республиканских
изданиях', 60, null),
(2023, 'Опубликованная научная монография в зарубежных изданиях', 80, null),
(2024, 'Опубликованная научная монография в дальнем зарубежье', 125, null),
(2025, 'Диплом на открытие', 150, null),
(2026, 'Инновационный патент, патент, свидетельство на авторское право
зарегистрированный в Казахстане', 35, null),
(2027, 'Инновационный патент, патент, свидетельство на авторское право
зарегистрированный в России', 60, null),
```


(2028, 'Инновационный патент, патент, свидетельство на авторское право зарегистрированный в международных органах по фиксации интеллектуальной собственности', 75, null),
(2029, 'Статья, опубликованная в республиканских изданиях, не входящих в Перечень ККОН (ВАК) РК', 10, null),
(2030, 'Статья, опубликованная в республиканских изданиях, входящих в Перечень ККОН (ВАК) РК', 20, null),
(2031, 'Статья, опубликованная в странах ближнего зарубежья', 25, null),
(2032, 'Статья, опубликованная в странах дальнего зарубежья', 30, null),
(2033, 'За каждую ссылку в республиканских изданиях', 3, null),
(2034, 'За каждую ссылку на статью в международных изданиях', 8, null),
(2035, 'Научный доклад сделанный на научной конференции, симпозиуме, конгрессе – республиканский уровень', 10, null),
(2036, 'Научный доклад сделанный на научной конференции, симпозиуме, конгрессе – международный уровень', 20, null),
(2037, 'Научный доклад сделанный на научной конференции, симпозиуме, конгрессе – дальнее зарубежье', 30, null),
(2038, 'Работа в секциях научно-технических и научно-методических советов и комиссий при органах управления высшей школой, министерств и ведомств', 40, null),
(2039, 'Участие в работе конкурсных научно-технических комиссиях – факультетские', 5, null),
(2040, 'Участие в работе конкурсных научно-технических комиссиях – институтские', 15, null),
(2041, 'Руководство научным семинаром (кружком) студентов (с периодичностью не менее 2-х раз в месяц), группой не менее 5 студентов', 25, null),
(2042, 'Руководство НИРС группой до 4 студентов', 20, null),
(2043, 'Руководство призерами факультетских туров НИРС', 20, null),
(2044, 'Руководство призерами институтских туров', 35, null),
(2045, 'Руководство призерами республиканских туров', 45, null),
(2046, 'Руководство призерами международных конкурсов', 80, null),
(2047, 'Руководство выпускными работами бакалавров – призерами республиканских и институтских конкурсов', 45, null)

4.2.8 Заполнение таблицы Преподаватель

```
insertintoPrepodschema.prepod
values (620001, 'Куралбаев', 'Зауытбек', 'Куралбаевич', 87773570774, 'Д.Ф.М.Н.',
'профессор', 620001, '22.04.1946', 3998602, 'Богенбай батыр', '187', 8001),
(620002, 'Бимурзаев', 'Сеиткерим', 'Бимурзаевич', 87014651014, 'Д.Ф.М.Н.',
'профессор', 620001, '05.11.1942', 3761171, 'Нурмакова', '81', 8001),
(620003, 'Ташимов', 'Малай', 'Алькинич', null, 'Д.Т.Н.',
'профессор', 620001, '20.05.1934', null, 'ул. Валиханова', '11', 8001),
(620004, 'Шанаев', 'Орынгали', 'Толегенович', 87772445677, 'К.П.Н.',
'профессор', 620001, '10.03.1943', 2272975, 'Токтабаев', '11', 8001),
(620005, 'Сыдыков', 'Бакыт', 'Диханбаевич', 87076157377, 'Д.П.Н.',
'доцент', 620001, '07.05.1968', null, 'Варшавская', '47/7', 8001),
(620006, 'Шайхин', 'Берк', 'Мурзахметович', 87055782194,
'К.Ф-М.Н.', 'доцент', 620001, '23.06.1942', 2203878, 'Орбита 3', '30', 8001),
(620007, 'Жунусов', 'Зангар', 'Абильхасимович', 87012122054, 'К.Т.Н.',
'доцент', 620001, '07.08.1947', 2295698, 'Щепкина', '37', 8001),
(620008, 'Ахметова', 'Майра', 'Ахметовна', 87014813960, 'К.Т.Н.',
'доцент', 620001, '14.12.1949', null, 'Толе-би', '221', 8001),
(620010, 'Турганбаев', 'Ерик', 'Сулейменович', 87052334228,
'К.Ф-М.Н.', 'доцент', 620001, '25.10.1959', 2410749, 'Толе-би', '261', 8001),
(620011, 'Коржымбаев', 'Турсын', 'Толебаевич', 87772268359,
'К.Ф-М.Н.', 'доцент', 620001, '10.12.1942', null, 'Таугул 2', '30', 8001),
(620012, 'Аманбаев', 'Абдирахман', 'Абдиханович', 87017761527, 'К.Ф-М.Н.',
'Ст. Преподаватель', 620001, '22.11.1960', 3847474, 'Торайгырова', '45', 8001),
(620013, 'Сатимова', 'Елена', 'Григорьевна', 87013253555, 'К.Т.Н.',
```

'Ст.
 Преподаватель', 620001, '15.08.1957', 2715651, 'Минусинская', '23/17', 8001),
 (620014, 'Жантлеуов', 'Кенжебек', 'Кожанович', 87073442107, 'К.Т.Н.',
 'Ст. Преподаватель', 620001, '23.06.1967', 2391427, 'Римова', '9', 8001),
 (620015, 'Сербин', 'Василий', 'Валерьевич', 87057777946, 'К.Т.Н.',
 'Ст. Преподаватель', 620001, '03.07.1984', 934446, 'Алмагуль', '27', 8001),
 (620016, 'Кожамбердиева', 'Мергул', 'Иманбековна', 87074809917, 'К.П.Н.',
 'Доцент', 620001, '02.06.1967', 3096649, 'Пятницкая', '81', 8001),
 (620017, 'Мусапирова', 'Гульзада', 'Даулетбековна', 87773778479, 'К.Т.Н.',
 'Ст. Преподаватель', 620001, '18.03.1982', 3098668, 'Жетісу 3', '66', 8001),
 (620018, 'Мусатаева', 'Гульсим', 'Толегеновна', 87013976463, null,
 'Ст. Преподаватель', 620001, '01.03.1969', 2236477, 'мкр. Аксай 2', '34', 8001),
 (620019, 'Нурмагамбетов', 'Галымжан', 'Саруарович', 87775858266, null,
 'Ст. Преподаватель', 620001, '01.01.1950', 3848604, 'Жангильдина', '147', 8001),
 (620020, 'Тойгожинова', 'Айнур', 'Жумакановна', 87013822300, null,
 'Ст. Преподаватель', 620001, '19.01.1978', 2524819, 'Айнабулак
 3', '125/29', 8001),
 (620021, 'Ержан', 'Асель', 'Ануаровна', 87022101147, null, 'Ст.
 Преподаватель', 620001, '22.07.1981', 2926682, 'Мечникова', '97/60', 8001),
 (620022, 'Конуспаева', 'Айгуль', 'Толегеновна', 87775140640, null, 'Ст.
 Преподаватель', 620001, '02. 09.1970', 226477, 'мкр. Аксай 2', '34/ 52', 8001),
 (620023, 'Байжанова', 'Дина', 'Ондасыновна', 87777917602, null, 'Ст.
 Преподаватель', 620001, '26.02.1979', null, 'Богенбай батыра', '174', 8001),
 (620024, 'Зуева', 'Екатерина', 'Александровна', 87772339848, null, 'Ст.
 Преподаватель', 620001, '23.07.1981', 2626558, 'Самал-2', '21', 8001),
 (620025, 'Рахимжанова', 'Зухра', 'Муратовна', 87059876525, null, 'Ст.
 Преподаватель', 620001, '31.05.1979', 2606577, 'Наурызбай батыра', '148', 8001),
 (620026, 'Тергеусизова', 'Алия', 'Советжановна', 87052401440, null, 'Ст.
 Преподаватель', 620001, '10.01.1982', 825233912, 'м-н Водник 1', '51', 8001),
 (620027, 'Ельчибаева', 'Райгуль', 'Балабаевна', 87014578016, null, 'Ст.
 Преподаватель', 620001, '17.03.1980', 2517185, 'Садовая', '7', 8001),
 (620028, 'Таурбекова', 'Айнур', 'Адилгазыевна', 87077181003, null, 'Ст.
 Преподаватель', 620001, '23.07.1971', null, 'Ул.алатау', '18', 8001),
 (620029, 'Оразаева', 'Айнур', 'Ришатовна', 87753946490, null, 'Ст.
 Преподаватель', 620001, '14.11.1983', null, 'ул.Айманова', '124', 8001),
 (620030, 'Жомартов', 'Манат', 'Абилович', 87029780274, null, 'Ст.
 Преподаватель', 620001, '01.09.1962', 2761208, 'Нурмакова', '81', 8001),
 (620031, 'Тлеубаева', 'Айсара', 'Жалгасбековна', 87773967289, null,
 'Ассистент', 620001, '04.02.1981', 87275233765, 'м-н Водник 1', '48', 8001),
 (620032, 'Акижанова', 'Зауре', 'Абиевна', 87004440666, null,
 'Ассистент', 620001, '21.06.1948', 3869717, 'Касымова', '22', 8001),
 (620033, 'Чуматанова', 'Шынар', 'Баймуратовна', 87021774482, null,
 'Ассистент', 620001, '17.05.1977', null, 'ул.Шугла', '52', 8001),
 (620034, 'Тусупов', 'Даулет', 'Мухамедович', 87772824777, null,
 'Ассистент 0,5', 620001, '01.01.1969', 2262317, 'Мамыр 4', '308', 8001),
 (620035, 'Карымсакова', 'Нургуль', 'Тлетаевна', null, null,
 'Ассистент', 620001, '18.10.1986', null, 'Жулдыз', '50', 8001),
 (620036, 'Ыбытаева', 'Галия', 'Сейткалиевна', null, null,
 'Ассистент', 620001, '03.08.1986', null, null, null, 8001),
 (620037, 'Сейлханова', 'Жанна', 'Адилхановна', null, null,
 'Ассистент 0,5', 620001, '01.01.1989', null, null, null, 8001),
 (620038, 'Оспищев', 'Максим', 'Валерьевич', null, null,
 'Ассистент 0,5', 620001, '16.12.1988', null, null, null, 8001),
 (620009, 'Абдрахманов', 'Темиржан', 'Еркешович', null, null,
 'Ассистент 0,5', 620001, '12.08.1989', null, null, null, 8001);

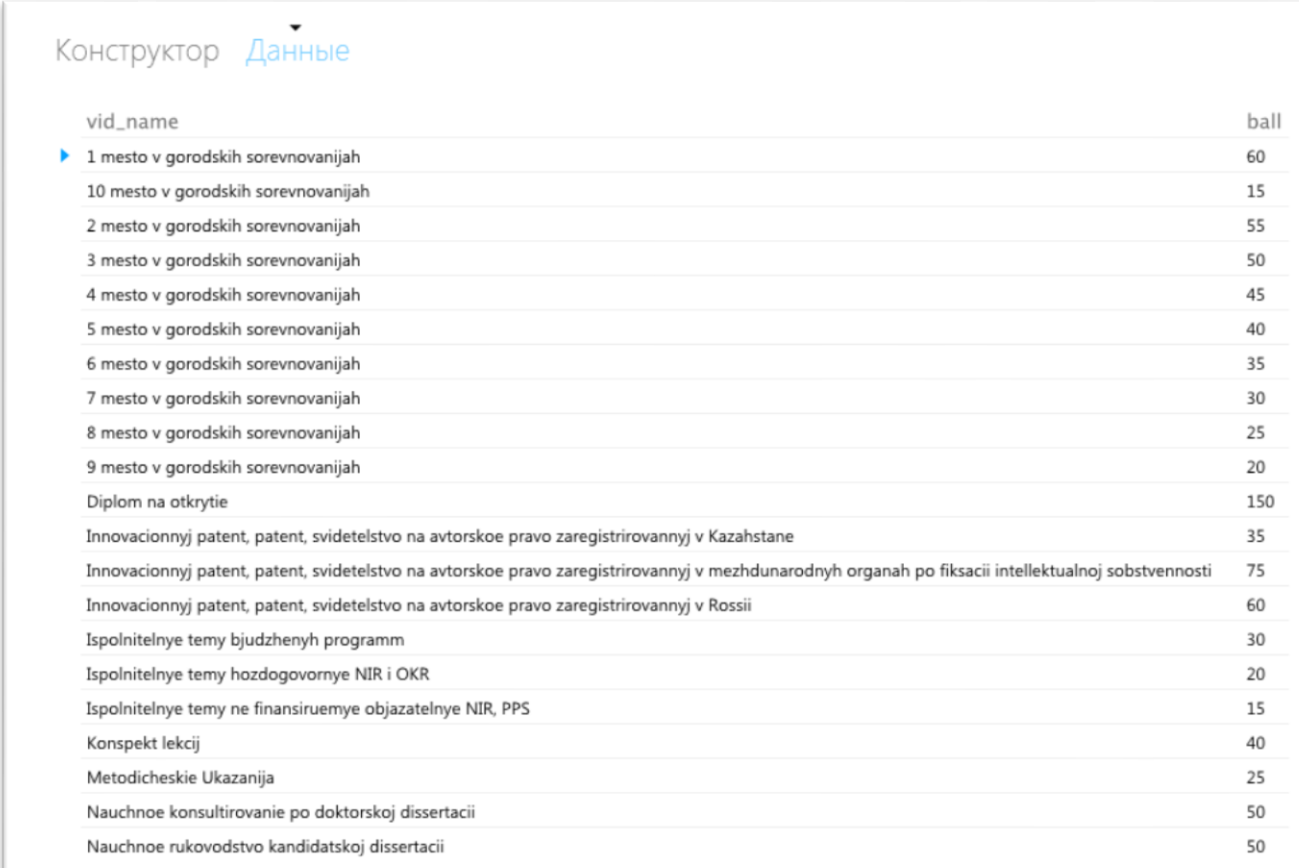
4.3 Запросы базы данных. Представления, процедуры и триггеры

4.3.1 Представления

Для запроса информации из БД были созданы специальные представления.

1 Служит для представления информации о видах работ и видах физ. воспитания. Результат работы представления «vidi» представлен на рисунке 4.2.

```
createviewvidias  
(selectvid_name,ball  
fromvid  
union  
selectvid_name,ball  
fromvid_fiz)
```



Конструктор Данные

| vid_name | ball |
|---|------|
| ▶ 1 место v gorodskih sorevnovanijah | 60 |
| 10 место v gorodskih sorevnovanijah | 15 |
| 2 место v gorodskih sorevnovanijah | 55 |
| 3 место v gorodskih sorevnovanijah | 50 |
| 4 место v gorodskih sorevnovanijah | 45 |
| 5 место v gorodskih sorevnovanijah | 40 |
| 6 место v gorodskih sorevnovanijah | 35 |
| 7 место v gorodskih sorevnovanijah | 30 |
| 8 место v gorodskih sorevnovanijah | 25 |
| 9 место v gorodskih sorevnovanijah | 20 |
| Diplom na otkrytie | 150 |
| Innovacionnyj patent, patent, svidetelstvo na avtorskoe pravo zaregistririvannyj v Kazahstane | 35 |
| Innovacionnyj patent, patent, svidetelstvo na avtorskoe pravo zaregistririvannyj v mezhdunarodnyh organah po fiksacii intellektualnoj sobstvennosti | 75 |
| Innovacionnyj patent, patent, svidetelstvo na avtorskoe pravo zaregistririvannyj v Rossii | 60 |
| Ispolnitelnye temy bjudzhenyh programm | 30 |
| Ispolnitelnye temy hozdogovornye NIR i OKR | 20 |
| Ispolnitelnye temy ne finansiruemye objazatelnye NIR, PPS | 15 |
| Konspekt lekcij | 40 |
| Metodicheskie Ukazanija | 25 |
| Nauchnoe konsultirovanie po doktorskoj dissertacii | 50 |
| Nauchnoe rukovodstvo kandidatskoj dissertacii | 50 |

Рисунок 4.2 – Результат работы представления «vidi»

2 Служит для отображения имени и фамилии преподавателя, кафедры, на которой он работает и общего коэффициента. Результат работы представления «ASS1» представлен на рисунке 4.3.

```

createviewASS1as
(selectprepod_fam,prepod_imya,prepod_otch,doljnost,kafedra_name,ob_kef
fromPrepodschema.prepodp
joinPrepodschema.kafedrak
onk.kafedra_id=p.kafedra_id
joinkeff
onf.prepod_id=p.prepod_id)

```

Конструктор [Данные](#)

| prepod_fam | prepod_imya | prepod_otch | doljnost | kafedra_name | ob_kef |
|----------------|-------------|----------------|------------------|--------------|--------|
| ▶ Kuralbaev | Zauytbek | Kuralbaevich | professor | Kafedra KT | 0 |
| Bimurzaev | Seitkerim | Bimurzaevich | professor | Kafedra KT | 6 |
| Tashimov | Malay | Alkinich | professor | Kafedra KT | 0 |
| Shanaev | Oryngali | Tolegenovich | professor | Kafedra KT | 21 |
| Sydykov | Bakyt | Dihanbaevich | dotsent | Kafedra KT | 0 |
| Shayhin | Berk | Murzahmetovich | dotsent | Kafedra KT | 5 |
| Zhunusov | Zangar | Abilhasimovich | dotsent | Kafedra KT | 4 |
| Ahmetova | Mayra | Ahmetovna | dotsent | Kafedra KT | 3 |
| Abdrahmanov | Temirzhan | Erkeshovich | Assistent 0,5 | Kafedra KT | 0 |
| Turganbaev | Eruk | Suleimenivich | dotsent | Kafedra KT | 4 |
| Korzhymbaev | Tursyn | Tolebaevich | dotsent | Kafedra KT | 0 |
| Amanbaev | Abdirahman | Abdihanovich | St. Prepodavatel | Kafedra KT | 3 |
| Satimova | Elena | Grigorevna | St. Prepodavatel | Kafedra KT | 1 |
| Zhantleuov | Kenzhebek | Kozhanovich | St. Prepodavatel | Kafedra KT | 0 |
| Serbin | Vasiliy | Valerevich | St. Prepodavatel | Kafedra KT | 0 |
| Kozhamberdieva | Mergul | Imanbekovna | Dotsent | Kafedra KT | 0 |
| Musapirova | Gulzada | Dauletbekovna | St. Prepodavatel | Kafedra KT | 0 |
| Musataeva | Gulsim | Tolegenovna | St. Prepodavatel | Kafedra KT | 16 |
| Nurmagambetov | Galymzhan | Saruarovich | St. Prepodavatel | Kafedra KT | 0 |
| Toygozhinova | Aynur | Zhumakanovna | St. Prepodavatel | Kafedra KT | 5 |
| Erzhan | Asel | Anuarovna | St. Prepodavatel | Kafedra KT | 9 |

Рисунок 4.3 – Результат работы представления «ASS1»

3 Служит для отображения информации о преподавателях. Результат работы представления «prepod1» представлен на рисунке 4.4.

```

Create viewPrepodias
(selectprepod_fam,prepod_imya,prepod_otch,prepod_tel,u_stepen,doljnost,dat
a,dom_tel,adress,dom,kafedra_name
fromPrepodschema.prepodp
joinPrepodschema.kafedrak
onp.kafedra_id=p.kafedra_id)

```

Конструктор [Данные](#)

| prepod_fam | prepod_imya | prepod_otch | prepod_tel | u_stepen | doljnost | kafedra_name |
|----------------|-------------|----------------|-------------|-----------|------------------|--------------|
| ▶ Kuralbaev | Zauytbek | Kuralbaevich | 87773570774 | D.F.M.N. | professor | Kafedra KT |
| Bimurzaev | Seitkerim | Bimurzaevich | 87014651014 | D.F.M.N. | professor | Kafedra KT |
| Tashimov | Malay | Alkinich | <null> | D.T.N. | professor | Kafedra KT |
| Shanaev | Oryngali | Tolegenovich | 87772445677 | K.P.N. | professor | Kafedra KT |
| Sydykov | Bakyt | Dihanbaevich | 87076157377 | D.P.N. | dotsent | Kafedra KT |
| Shayhin | Berk | Murzahmetovich | 87055782194 | K.F.-M.N. | dotsent | Kafedra KT |
| Zhunusov | Zangar | Abilhasimovich | 87012122054 | K.T.N. | dotsent | Kafedra KT |
| Ahmetova | Mayra | Ahmetovna | 87014813960 | K.T.N. | dotsent | Kafedra KT |
| Abdrahmanov | Temirzhan | Erkeshovich | <null> | <null> | Assistent 0,5 | Kafedra KT |
| Turganbaev | Eruk | Suleimenivich | 87052334228 | K.F.-M.N. | dotsent | Kafedra KT |
| Korzhymbaev | Tursyn | Tolebaevich | 87772268359 | K.F.-M.N. | dotsent | Kafedra KT |
| Amanbaev | Abdirahman | Abdihanovich | 87017761527 | K.F.-M.N. | St. Prepodavatel | Kafedra KT |
| Satimova | Elena | Grigorevna | 87013253555 | K.T.N. | St. Prepodavatel | Kafedra KT |
| Zhantleuov | Kenzhebek | Kozhanovich | 87073442107 | K.T.N. | St. Prepodavatel | Kafedra KT |
| Serbin | Vasily | Valerevich | 87057777946 | K.T.N. | St. Prepodavatel | Kafedra KT |
| Kozhambardieva | Mergul | Imanbekovna | 87074809917 | K.P.N. | Dotsent | Kafedra KT |
| Musapirova | Gulzada | Dauletbekovna | 87773778479 | K.T.N. | St. Prepodavatel | Kafedra KT |
| Musataeva | Gulsim | Tolegenovna | 87013976463 | <null> | St. Prepodavatel | Kafedra KT |
| Nurmagambetov | Galymzhan | Saruarovich | 87775858266 | <null> | St. Prepodavatel | Kafedra KT |
| Toygozhinova | Aynur | Zhumakanovna | 87013822300 | <null> | St. Prepodavatel | Kafedra KT |
| Frzhan | Acel | Anuarovna | 87022101147 | <null> | St. Prepodavatel | Kafedra KT |

Рисунок 4.4 – Результат работы представления «prepod_i»

4 Служит для отображения преподавателей и работ, которые они написали. Результат работы представления «raboti» представлен на рисунке 4.5.

```
createviewrabortias
(selectp.prepod_fam,m.nazvanie
fromPrepodschema.prepodp
joinavtorstvoa
ona.prepod_id=p.prepod_id
joinrabotar
onr.rabota_id=a.rabota_id
joinPrepodschema.metodm
onm.rabota_id=r.rabota_id
union
selectp.prepod_fam,n.nauch_name
fromPrepodschema.prepodp
joinavtorstvoaona.prepod_id=p.prepod_id
joinrabotar
onr.rabota_id=a.rabota_id
joinPrepodschema.nauchravn
onn.rabota_id=r.rabota_id)
```

4.3.2 Процедуры

1 Служит для добавления новой кафедры. Результат выполнения процедуры «new_kaf» представлен на рисунке 4.6.

Конструктор Данные

| prepod_fam | nazvanie |
|------------|--|
| ▶ Ahmetova | Electromagnetic mirror objective with removable spherical aberration |
| Akizhanova | Informatika |
| Akizhanova | Konspekt lekcii |
| Akizhanova | Postanovka 4 lab. rabot |
| Akizhanova | Programmirovaniye v DELPHI |
| Amanbaev | Elektromagnitnoye transaksialnoye zerkalo |
| Amanbaev | Postanovka 3 lab rabot po OMPT |
| Bayzhanova | Komputernie seti MU k kurs. rabotam rus 2011 |
| Bayzhanova | Komuternie seti MU k lab. rabotam kaz 2011 |
| Bayzhanova | Ladder filters with converters of resistance, Germany |
| Bayzhanova | Organizatsiya sistemi teh obslujivaniya |
| Bayzhanova | Ossobennosti teh sodержaniya gruzovih avtomobilei |
| Bayzhanova | Postanovka 8 lab. rabot |
| Bayzhanova | Postanovka lab rabot |
| Bayzhanova | Problema razvitiya avtomobilnogo rinka RK |
| Bayzhanova | Programmirovaniye v DELPHI |
| Bayzhanova | Proizvodstvennaya praktika spec. VTiPO |
| Bayzhanova | Rabota v Microsoft SQL Server 2008 |
| Bayzhanova | RUP po IS i VTiPO |
| Bimurzaev | Poviwenie kachestva IT obrazovaniya |
| Bimurzaev | Raschet parametrov besprovodnogo kanala |

Рисунок 4.5 – Результат работы представления «raboti»

```

createprocedurenew_kaf(
@kaf_name  varchar(18))
as
declare@kaf_id  int;
select@kaf_id=(selectmax(kafedra_id)+1 fromPrepodschema.kafedra)
insertintoPrepodschema.kafedra
values (@kaf_id,@kaf_name);

execnew_kaf 'Kafedra Toe';

```

| kafedra_id | kafedra_name |
|------------|--------------|
| 8001 | Kafedra KT |
| 8002 | Kafedra Toe |

+ Добавить строку - Удалить строку

Рисунок 4.6 – Результат выполнения процедуры «new_kaf»

2 Служит для добавления нового аида анкет. Результат выполнения процедуры «new_vid_anket» представлен на рисунке 4.7.

```
create procedure new_vid_anket (
  @name varchar(90)
)
as
declare @id int
select @id = (select max(vid_ocenki) + 1 from vid_anket)
insert into vid_anket
values (@id, @name);

exec new_vid_anket 'Анкетирование Ректором'
```

| vid_ocenki | vid_name |
|------------|--------------------------------|
| 1 | Anketirovanie studentami |
| 2 | Anketirovanie prepodavateljami |
| 3 | Anketirovanie Rektorom |

+ Добавить строку - Удалить строку

Рисунок 4.7 – Результат выполнения процедуры «new_vid_anket»

4.3.3 Триггеры

1 Триггер для расчета коэффициента К1 и К3.

```
create trigger K1K3
on avtorstvo
for insert, update
as
declare @rabota_min int;
declare @rabota_max int;
declare @vid int;
declare @prepod int;
declare @prepod_1 int;
declare @ball int;
declare @ball_2 int;
declare @ball_x int;
```

```

declare@shtrafint;
declare@data_1datetime;
declare@data_2datetime;
declare@xint;
declare@yint;
select@ball_2=0
select@prepod=(selecti.prepod_idfrominsertedi)
select@rabota_min=(selectmin(i.rabota_id)frominsertediwherei.prepod_id=@prepod)
select@rabota_max=(selectmax(i.rabota_id)frominsertediwherei.prepod_id=@prepod)
while@rabota_min<=@rabota_max
begin
set@ball_x=0
if@rabota_min=some(selectrabota_idfromPrepodschema.metod)
begin
print(@rabota_min)
select@prepod_1=0
select@data_2=0
select@vid=(selecti.vid_idfromrabotaiwherei.rabota_id=@rabota_min)
select@ball=(selecti.ballfromvidiwherei.vid_id=@vid)
select@prepod_1=(selectprepod_idfrominsertedwhererabota_id=@rabota_minandprepod_id<>@prepod)
select@data_1=(selecti.data_sdachifromPrepodschema.metodiwherei.rabota_id=@rabota_min)
select@data_2=(selecti.data_planfromPrepodschema.metodiwherei.rabota_id=@rabota_min)
select@shtraf=(selectshtrafffromvidiwherevid_id=@vid)
if(year(GETDATE()-@data_1)-1900)<3
begin
set@ball_x=1
print(year(GETDATE()-@data_1)-1900)
if@prepod_1>0 set@x=1 elseset@x=0
if@data_2>0 set@y=1 elseset@y=0
if@x=0
begin
if@y=1
begin
if@data_2-@data_1>0
begin
set@ball_2+=@ball
print(@ball_2)
end
if@data_1-@data_2>0
begin
set@ball_2+=(@ball-@shtraf)
print(@ball_2)
end
end
if@y=0
begin
set@ball_2+=@ball
print(@ball_2)
end
end
if@x=1
begin
if@y=1
begin
if@data_2-@data_1>0
begin
set@ball_2+=(@ball/2)
updatekef

```



```

setkef_1-= (@ball/2)
whereprepod_id=@prepod_1
end
if@data_1-@data_2>0
begin
set@ball_2+= ((@ball-@shtraf)/2)
updatekef
setkef_1-= ((@ball-@shtraf)/2)
end
end
if@y=0
begin
set@ball_2+=@ball/2
print (@ball_2)
end
end
end
updatekef
setkef_1+= (@ball_x+ (@ball_2/20))
whereprepod_id=@prepod
print (@ball_2)
print (@ball_x+ (@ball_2/20))
end
if@rabota_min=some (selectrabota_idfromPrepodschema.nauchrab)
begin
print (@rabota_min)
select@prepod_1=0
select@vid=(selecti.vid_idfromrabotaiwherei.rabota_id=@rabota_min)
select@ball=(selecti.ballfromvidiwherei.vid_id=@vid)
select@prepod_1=(selectprepod_idfrominsertedwhererabota_id=@rabota_minandp
repop_id<>@prepod)
select@data_1=(selecti.data_sdachifromPrepodschema.nauchrabiwherei.rabota_
id=@rabota_min)
print ('*****')
if (year (GETDATE ())-@data_1)-1900 <2
begin
set@ball_x=1
print (year (GETDATE ())-@data_1)-1900)
if@prepod_1>0 set@x=1 elseset@x=0
if@x=0
begin
set@ball_2+=@ball
print (@ball_2)
end
if@x=1
begin
set@ball_2+=@ball/2
print (@ball_2)
end
end
updatekef
setkef_3+= (@ball_x+ (@ball_2/20))
whereprepod_id=@prepod
print (@ball_2)
print (@ball_x+ (@ball_2/20))
end
select@rabota_min=(selectmin (i.rabota_id) frominsertediwherei.rabota_id>@ra
bota_min)
end

```

2 Триггер для расчета коэффициента К2.

```
        createtriggerK2
onocenaschema.anketirovanie
forinsert,update
as
declare@ballint;
declare@prepodint;
select@prepod=(selectprepod_idfrominsertedi)
select@ball=(selecti.ballifrominsertediwherei.vid_ocenki=2
andprepod_id=@prepod)
updatekef
setkef_2=(@ball/20)
whereprepod_id=@prepod
```

3 Триггер для расчета коэффициента К6.

```
createtriggerK6
onfiz
forinsert,update
as
declare@ballint;
declare@prepodbigint;
declare@vidint;
select@prepod=(selectprepod_idfrominserted)
select@vid=(selectvid_idfrominsertedwhereprepod_id=@prepod)
select@ball=(selectballfromvid_fizwherevid_id=@vid)
if@vid=2064 and@ball>30
begin
ROLLBACKTRAN
RAISERROR ('Максимальное количество баллов не должно превышать 30', 16,10)
end
updatekef
setkef_6=(@ball/20)
whereprepod_id=@prepod
```

4 Триггер для расчета коэффициента К4.

```
createtriggerK4
onocenki
forinsert,update
as
declare@prepodint;
declare@ballint;
select@prepod=(selectprepod_idfrominsertedwherevid_id=0001)
select@ball=(selectballfrominsertedwhereprepod_id=@prepod)
if@ball>2
begin
ROLLBACKTRAN
RAISERROR ('Максимальный балл устанавливается в диапазоне от 0 до 2',
16,10)
end
updatekef
setkef_4=@ball
whereprepod_id=@prepod
```

5 Триггер для расчета коэффициента K5.

```
createtriggerK5
onocenaschema.anketirovanie
forinsert,update
as
declare@ballfloat;
declare@prepodint;
select@prepod=(selectprepod_idfrominsertedi)
select@ball=(SELECTROUND(i.balli, 1)frominsertediwherei.vid_ocenki=1
andprepod_id=@prepod)
if@ball>2.9
begin
updatekef
setkef_5=@ball
whereprepod_id=@prepod
print('*****')
print(@ball)
print('*****')
end
```

6 Триггер для расчета коэффициента K7.

```
createtriggerK7
onocenki
forinsert,update
as
declare@prepodint;
declare@ballfloat;
select@prepod=(selectprepod_idfrominsertedwherevid_id=2)
select@ball=(selectballfrominsertedwhereprepod_id=@prepod)
if@ball>2.0
begin
ROLLBACKTRAN
RAISERROR ('Максимальный балл устанавливается в диапазоне от 0 до 2',
16,10)
end
if@ball-round(@ball,1)<>5 and@ball-round(@ball,1)<>0
begin
ROLLBACKTRAN
RAISERROR ('Диапазон изменений коэффициента устанавливается с шагом 0,5',
16,10)
end
updatekef
setkef_7=@ball
whereprepod_id=@prepod
print('*****')
```

7 Триггер для расчета общего коэффициента.

```
createtriggerOb_kef
onkef
forinsert,update
as
declare@prepodbigint;
declare@kef_1float;
declare@kef_2float;
declare@kef_3float;
declare@kef_4float;
```

```

declare@kef_5float;
declare@kef_6float;
declare@kef_7float;
declare@countint;
select@count=(selectcount(prepod_id)fromkef)
while@count>=0
begin
select@kef_1=0
select@kef_2=0
select@kef_3=0
select@kef_4=0
select@kef_5=0
select@kef_6=0
select@kef_7=0
select@prepod=(selectprepod_idfrominserted)
select@kef_1=(selectkef_1frominsertedwhereprepod_id=@prepod)
select@kef_2=(selectkef_2frominsertedwhereprepod_id=@prepod)
select@kef_3=(selectkef_3frominsertedwhereprepod_id=@prepod)
select@kef_4=(selectkef_4frominsertedwhereprepod_id=@prepod)
select@kef_5=(selectkef_5frominsertedwhereprepod_id=@prepod)
select@kef_6=(selectkef_6frominsertedwhereprepod_id=@prepod)
select@kef_7=(selectkef_7frominsertedwhereprepod_id=@prepod)
updatekef
setOb_kef=@kef_1+@kef_2+@kef_3+@kef_4+@kef_5+@kef_6+@kef_7
whereprepod_id=@prepod
set@count=@count-1
end

```

4.3.4 Функции

1 Служит для отображения оценок, выставленные деканатом. Результат выполнения функции «Get_ocenki» представлен на рисунке 4.8.

```

createfunctionGet_ocenki(@idbigint)
returnsint
as
begin
declare@nameint
select@name=(selectballfromocenkiwhereprepod_id=@idandball>0)
return@name
end

```

| Фамилия | Имя | Оценки |
|-----------|-----------|--------|
| Kuralbaev | Zauytbek | <null> |
| Bimurzaev | Seitkerim | <null> |
| Tashimov | Malay | <null> |
| Shanaev | Oyngali | 2 |
| Sydykov | Balyt | <null> |

Рисунок 4.8 – Результат выполнения функции «Get_ocenki»

1 Служит для отображения информации о количестве преподавателей на каждой кафедре. Результат выполнения функции «Get_kol_prepod» представлен на рисунке 4.9.

```
createfunctionget_kol_prepod(@idbigint,  
@namevarchar(30))  
returnsint  
as  
begin  
declare@countint  
select@count=(selectcount(prepod_id)fromPrepodschema.prepodpjoinPrepodsche  
ma.kafedrakonp.kafedra_id=k.kafedra_idwherekafedra_name=@namegroupbykafedr  
a_name)  
return@count  
end
```

```
select kafedra_name, dbo.get_kol_prepod(p.kafedra_name) as 'Количество'  
from prepodschema.kafedra p  
join prepodschema.prepod k  
on p.kafedra_id=p.kafedra_id  
group by kafedra_name
```

Сообщения Результаты

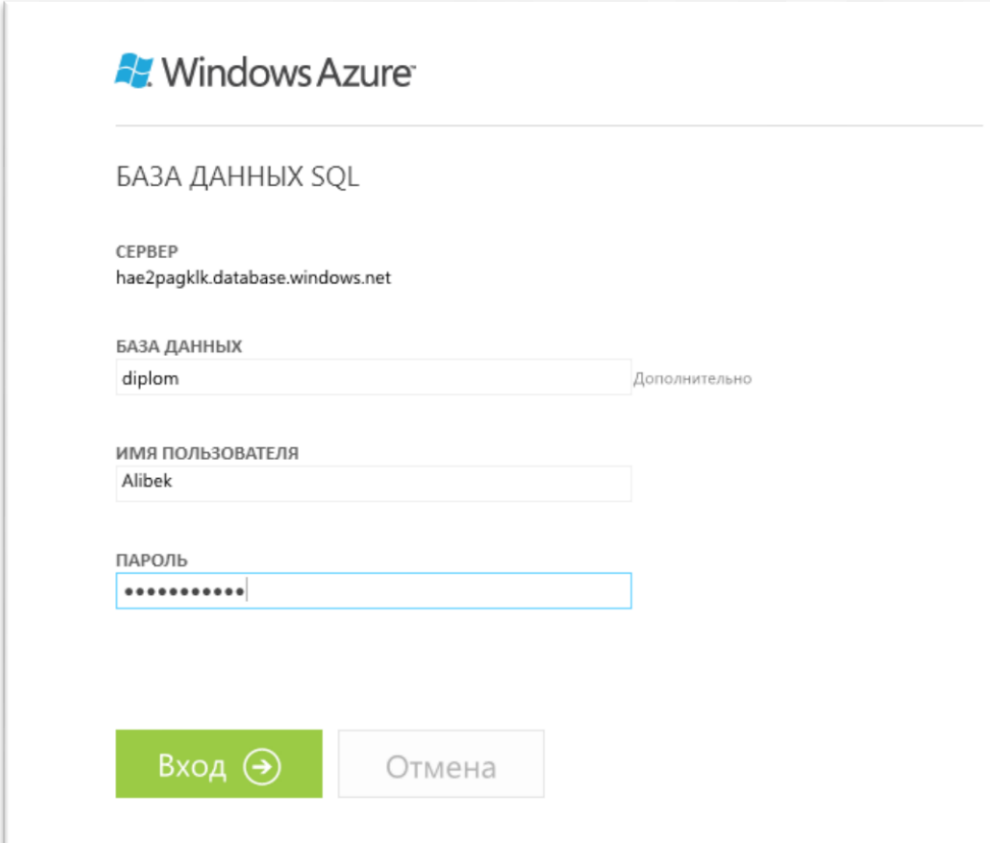
1 1 Строки

| kafedra_name | Количество |
|--------------|------------|
| ▶ Kafedra KT | 38 |

Рисунок 4.9 – Результат выполнения функции «Get_kol_prepod»

5 Описание интерфейса Windows Azure. Работа с приложением

Интерфейс данного приложения очень удобен для использования. С данным приложением могут работать несколько видов пользователей. По умолчанию создается один пользователь с правами администратора.



The screenshot shows the Windows Azure user authentication interface. At the top, the Windows Azure logo is displayed. Below it, the text "БАЗА ДАННЫХ SQL" (SQL DATABASE) is shown. Underneath, the server name "СЕРВЕР hae2pagkik.database.windows.net" is listed. The "БАЗА ДАННЫХ" (DATABASE) field contains the value "diplom" and has a "Дополнительно" (Advanced) link. The "ИМЯ ПОЛЬЗОВАТЕЛЯ" (USERNAME) field contains "Alibek". The "ПАРОЛЬ" (PASSWORD) field is masked with dots. At the bottom, there are two buttons: a green "Вход" (Login) button with a right arrow icon, and a grey "Отмена" (Cancel) button.

Рисунок 5.1 – Авторизация пользователя

Для начала работы с программой Вам, необходимо авторизоваться в системе и базе данных. Для этого:

- а) введите имя пользователя;
- б) введите пароль.

В случае неверно введенных данных вы получите сообщение, представленное на рисунке 5.2.

На форме находятся кнопки, при нажатии которых можно переходить на соответствующие формы для выполнения определенных операций. В верхней части располагаются формы, которые недавно открыли. Снизу располагаются основные формы. Меню форм показано на рисунке 5.3.

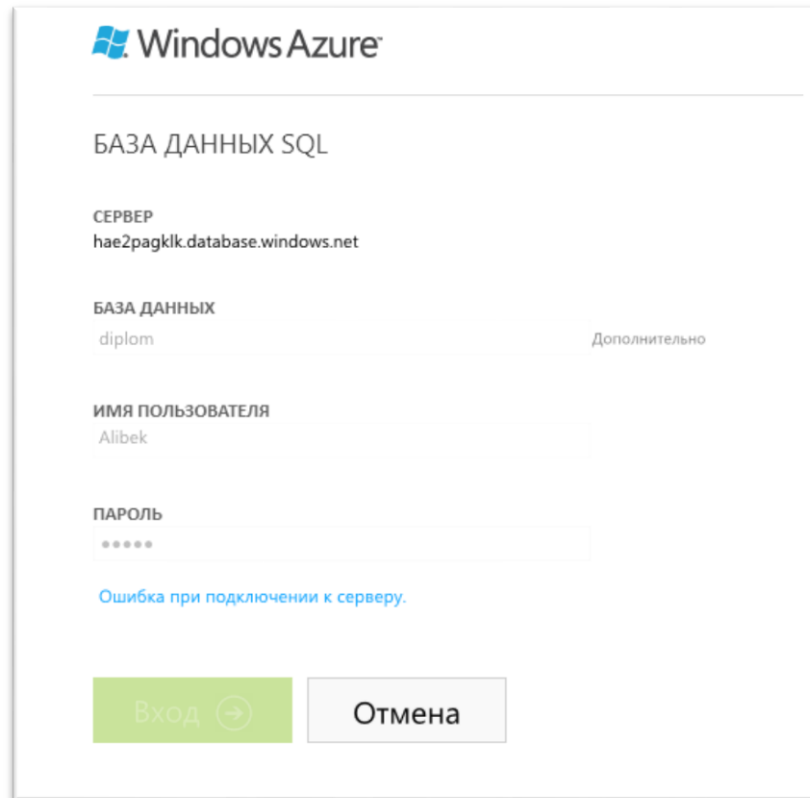


Рисунок 5.2 – Сообщение об ошибке при вводе данных

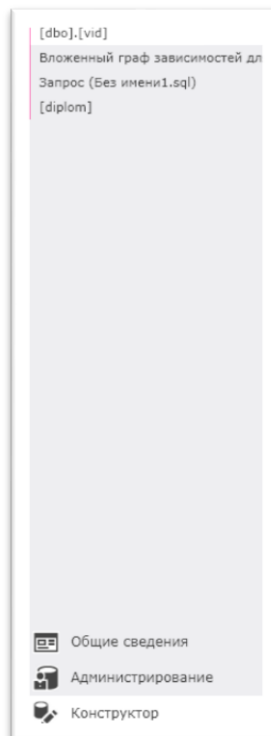


Рисунок 5.3 – Меню форм

При нажатии на кнопку «Общие сведения» система перенаправляет на форму, представленную на рисунке 5.4

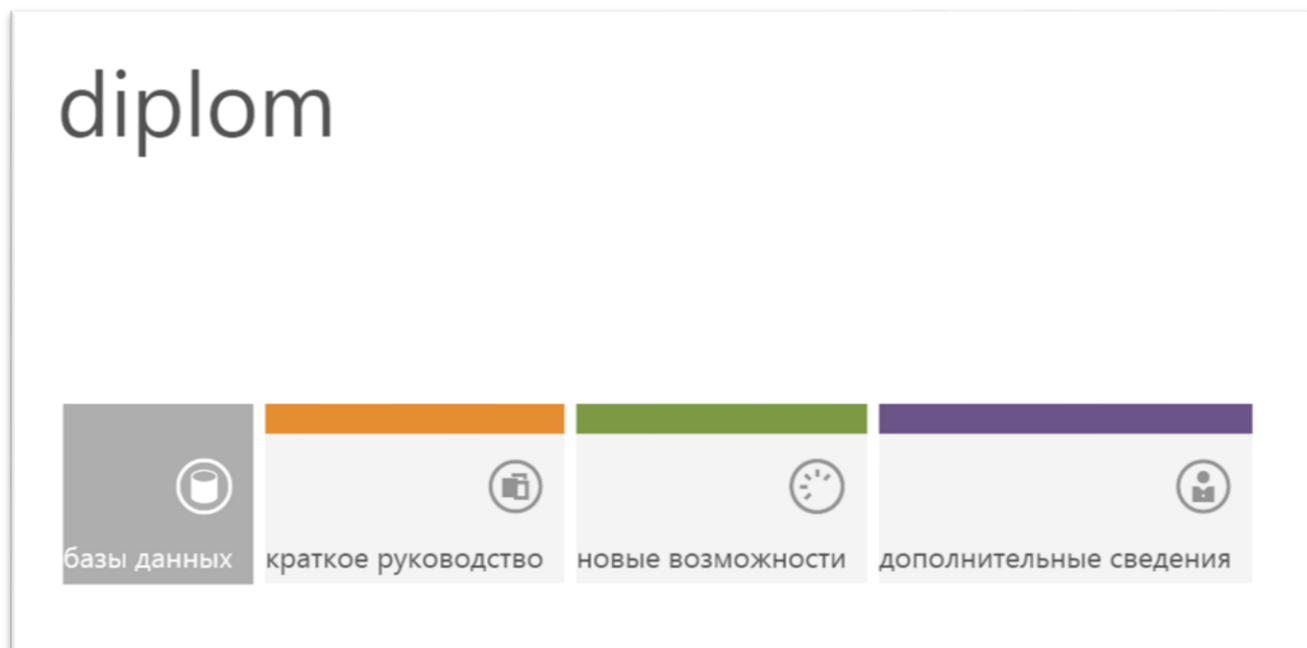


Рисунок 5.4 – Форма под названием «Общие сведения»

На форме, представленной на рисунке 5.4 пользователь имеет право просматривать данные о базе данных, руководство пользователя, возможности, которые пользователь может использовать и дополнительные сведения.

При нажатии формы «Администрирование» пользователь переходит на форму, представленную на рисунке 5.5.

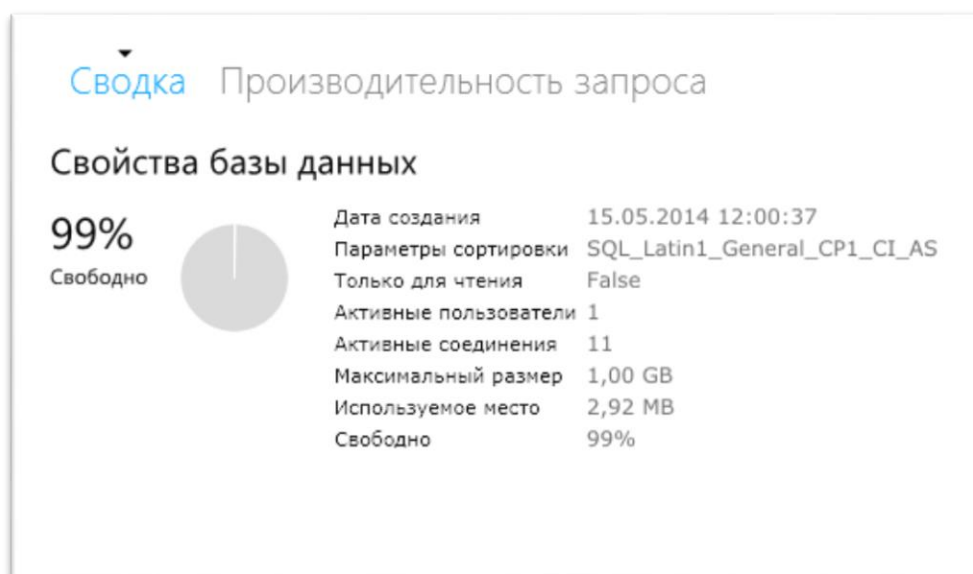


Рисунок 5.5 – Форма администратора

На форме, представленной на рисунке 5.5 пользователь может просматривать свойства базы данных и производительность запросов, которые показаны на рисунке 5.6.

Сводка **Производительность запроса**

| Запрос | Число запусков | ЦП мс/с | Длительность мс/с | Физические операции чтения | Логические операции записи | Логические операции чтения |
|--|----------------|---------|-------------------|----------------------------|----------------------------|----------------------------|
| SELECT CAST((select SUBSTRING(text,eq1.statement_start_offset/2 +1, ((CASE eq1 | 1 | 54 | 56 | 0 | 658 | 7611 |
| SELECT CAST((select SUBSTRING(text,eq1.statement_start_offset/2 +1, ((CASE eq1 | 1 | 52 | 52 | 0 | 672 | 7598 |
| SELECT CAST((select SUBSTRING(text,eq1.statement_start_offset/2 +1, ((CASE eq1 | 1 | 52 | 52 | 0 | 678 | 7524 |
| SELECT CAST((select SUBSTRING(text,eq1.statement_start_offset/2 +1, ((CASE eq1 | 1 | 22 | 22 | 0 | 0 | 76 |
| select prepod_fam as 'Фамилия', prepod_imya as 'Имя', dbo.get_ocenki(prepod_ic | 1 | 0 | 0 | 0 | 0 | 0 |
| SELECT database_id FROM sys.databases WHERE name=N'diplom' and db_name() | 48 | 0 | 0 | 0 | 0 | 0 |
| update kef set kef_1+=(@ball/20) where prepod_id=@@prepod | 47 | 0 | 0 | 0 | 0 | 0 |
| update kef set (tbl_1=tbl_2, tbl_2=tbl_3, tbl_3=tbl_4, tbl_4=tbl_5, tbl_5=tbl_6) | 0 | 0 | 0 | 0 | 0 | 0 |

Рисунок 5.6 – Форма под названием «Производительность запросов»

При нажатии на кнопку «Конструктор» система перенаправляет на форму, представленную на рисунке 5.7.

| Имя таблицы | Размер таблицы | Количество строк |
|------------------------|----------------|------------------|
| dbo.avtorstvo | 8,00 KB | 83 |
| dbo.dekanat | 8,00 KB | 1 |
| dbo.fiz | 0,00 KB | 0 |
| dbo.kef | 8,00 KB | 38 |
| dbo.ocenki | 8,00 KB | 3 |
| dbo.rabota | 8,00 KB | 53 |
| dbo.UP | 0,00 KB | 0 |
| dbo.vid | 8,00 KB | 47 |
| dbo.vid_anket | 8,00 KB | 3 |
| dbo.vid_fz | 8,00 KB | 16 |
| dbo.vid_ocenki | 8,00 KB | 2 |
| ocenki.schema | 0,00 KB | 0 |
| Prepod.schema.tbl_dia | 8,00 KB | 1 |
| Prepod.schema.metod | 8,00 KB | 28 |
| Prepod.schema.nauchrab | 8,00 KB | 24 |
| Prepod.schema.prepod | 8,00 KB | 38 |

Рисунок 5.7 – Форма под названием «Конструктор»

Форма, представленная на рисунке 5.7, является одной из самых важных форм при администрировании базы данных. В этой форме пользователь может изменять таблицы, данные в таблице, представления, процедуры, а также просматривать зависимости первого уровня и полностью граф зависимостей.

На рисунке 5.8 показан граф зависимости первого уровня.

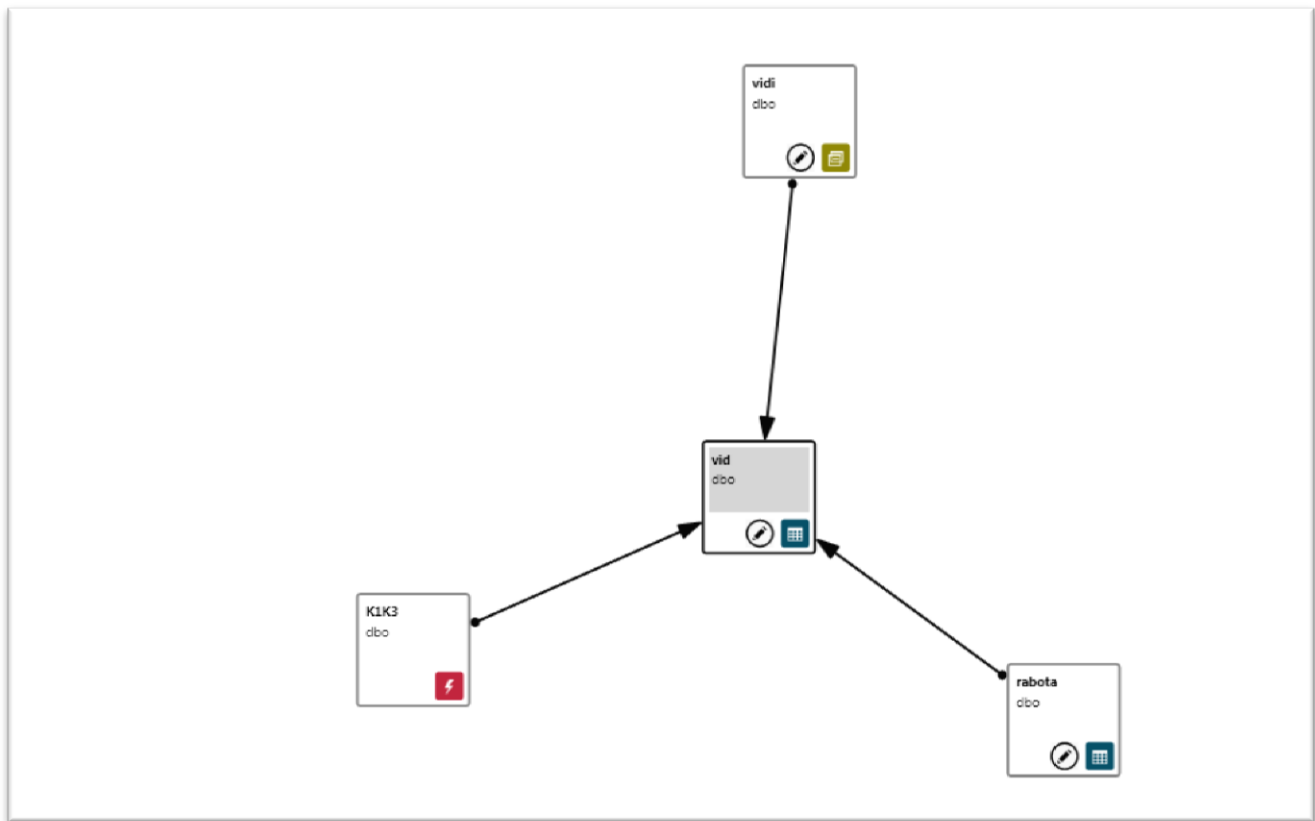


Рисунок 5.8 – Зависимости первого уровня для таблицы «Виды работ»

6 Безопасность жизнедеятельности

6.1 Анализ потенциально опасных и вредных факторов, влияющих на работу программиста

Работа с персональным компьютером – это воспроизведение визуальной информации на дисплее, которая должна быстро и точно восприниматься пользователем.

Основным фактором, влияющим на производительность труда людей, работающих с ПЭВМ и ВДТ, являются комфортные и безопасные условия труда.

Условия труда пользователя, работающего с персональным компьютером, определяются:

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

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

- 1) повышенная температура поверхностей ПК;
- 2) повышенная или пониженная температура воздуха рабочей зоны;
- 3) выделение в воздух рабочей зоны ряда химических веществ;
- 4) повышенная или пониженная влажность воздуха;
- 5) повышенный или пониженный уровень отрицательных и положительных аэроионов;
- 6) повышенное значение напряжения в электрической цепи, замыкание;
- 7) повышенный уровень статического электричества;
- 8) повышенный уровень электромагнитных излучений;
- 9) повышенная напряженность электрического поля;
- 10) отсутствие или недостаток естественного света;
- 11) недостаточная искусственная освещенность рабочей зоны;
- 12) повышенная яркость света;
- 13) повышенная контрастность;
- 14) прямая и отраженная блескость;
- 15) зрительное напряжение;
- 16) монотонность трудового процесса;
- 17) нервно–эмоциональные перегрузки.

Работа на ПК сопровождается постоянным и значительным напряжением функций зрительного анализатора. Одной из основных

особенностей является иной принцип чтения информации, чем при обычном чтении. При обычном чтении текст на бумаге, расположенный горизонтально на столе, считывается работником с наклоненной головой при падении светового потока на текст. При работе на ПК оператор считывает текст, почти не наклоняя голову, глаза смотрят прямо или почти прямо вперед, текст (источник – люминесцирующее вещество экрана) формируется по другую сторону экрана, поэтому пользователь не считывает отраженный текст, а смотрит непосредственно на источник света, что вынуждает глаза и орган зрения в целом работать в несвойственном ему стрессовом режиме длительное время.

Расстройство органов зрения резко увеличивается при работе более четырех часов в день. Всемирная организация здравоохранения (ВОЗ) ввела понятие “компьютерный зрительный синдром” (КЗС), типовыми симптомами которого являются жжение в глазах, покраснение век и конъюнктивы, чувство инородного тела или песка под веками, боли в области глазниц и лба, затуманивание зрения, замедленная перефокусировка с ближних объектов на дальние.

Нервно–эмоциональное напряжение при работе на ПК возникает вследствие дефицита времени, большого объема и плотности информации, особенностей диалогового режима общения человека и ПК, ответственности за безошибочность информации. Продолжительная работа на дисплее, особенно в диалоговом режиме, может привести к нервно–эмоциональному перенапряжению, нарушению сна, ухудшению состояния, снижению концентрации внимания и работоспособности, хронической головной боли, повышенной возбудимости нервной системы, депрессии.

Кроме того, при повышенных нервно–психических нагрузках в сочетании с другими вредными факторами происходит “выброс” из организма витаминов и минеральных веществ. При работе в условиях повышенных нервно–эмоциональных и физических нагрузок гиповитаминоз, недостаток микроэлементов и минеральных веществ (особенно железа, магния, селена) ускоряет и обостряет восприимчивость к воздействию вредных факторов окружающей и производственной среды, нарушает обмен веществ, ведет к изнашиванию и старению организма. Поэтому при постоянной работе на ПК для повышения работоспособности и сохранения здоровья к мерам безопасности относится защита организма с помощью витаминно–минеральных комплексов, которые рекомендуется применять всем, даже практически здоровым пользователям ПК.

Повышенные 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. Воспалительные процессы тканей сухожилий (тендениты) получили общее название “травма повторяющихся нагрузок”.

Большинство работающих рано или поздно начинают предъявлять жалобы на боли в шее и спине. Эти недомогания накапливаются постепенно и получили название “синдром длительных статических нагрузок” (СДСН).

Другой причиной возникновения СДСН может быть длительное пребывание в положении “сидя”, которое приводит к сильному перенапряжению мышц спины и ног, в результате чего возникают боли и неприятные ощущения в нижней части спины. Основной причиной перенапряжения мышц спины и ног являются нерациональная высота рабочей поверхности стола и сидения, отсутствие опорной спинки и подлокотников, неудобное размещение монитора, клавиатуры и документов, отсутствие подставки для ног.

Для существенного уменьшения боли и неприятных ощущений, возникающих у пользователей ПК, необходимы частые перерывы в работе и эргономические усовершенствования, в том числе оборудование рабочего места так, чтобы исключать неудобные позы и длительные напряжения.

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. Немаловажную роль играют эргономические параметры расположения экрана монитора (дисплея), состояние освещенности на рабочем месте, параметры мебели и характеристики помещения, где расположена компьютерная техника.

Требования Санитарных правил распространяются на вычислительные электронные цифровые машины персональные и портативные; периферические устройства вычислительных комплексов (принтеры, сканеры, клавиатуру, модемы внешние); устройства отображения информации (видеодисплейные терминалы – ВДТ) всех типов, условия и организацию работы с ПЭВМ и направлены на предотвращение неблагоприятного влияния на здоровье человека вредных факторов производственной среды и трудового процесса при работе с ПЭВМ.

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

Психофизические вредные и опасные факторы.

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.

Типичными ощущениями, которые испытывают к концу рабочего дня операторы ПЭВМ, являются: переутомление глаз, головная боль, тянущие боли в мышцах шеи, рук и спины, снижение концентрации внимания.

Уже в первые годы компьютеризации было отмечено специфическое зрительное утомление у пользователей дисплеев, получившее общее название «компьютерный зрительный синдром».

Одной из причин служит то, что сформировавшаяся за миллионы лет эволюции зрительная система человека приспособлена для восприятия объектов в отраженном свете (печатные тексты, рисунки и т.п.), а не для работы за дисплеем. Изображение на дисплее принципиально отличается от привычных глазу объектов наблюдения – оно светится, мерцает, состоит из дискретных точек, а цветное компьютерное изображение не соответствует естественным цветам. Но не только особенности изображения на экране вызывают зрительное утомление. Большую нагрузку орган зрения испытывает при вводе информации, так как пользователь вынужден часто переводить взгляд с экрана на текст и клавиатуру, находящиеся на разном расстоянии и по-разному освещенные. Зрительное утомление проявляется жалобами на затуманивание зрения, трудности при переносе взгляда с ближних предметов на дальние и с дальних на ближние, кажущиеся изменения окраски предметов, их двоение, чувство жжения, «песка» в глазах, покраснение век, боли при движении глаз.

Длительная и интенсивная работа на компьютере может стать источником тяжелых профессиональных заболеваний, таких, как травма повторяющихся нагрузок (ТПН), представляющая собой постепенно накапливающиеся недомогания, переходящие в заболевания нервов, мышц и сухожилий руки.

К профессиональным заболеваниям, связанным с ТПН, относятся:

- тендовагинит – воспаление сухожилий кисти, запястья, плеча;
- тендосиновит – воспаление синовиальной оболочки сухожильного основания кисти и запястья;
- синдром запястного канала (СЗК) – вызывается ущемлением срединного нерва в запястном канале. Накапливающаяся травма вызывает образование продуктов распада в области запястного канала, в результате чего вначале возникает отек, а затем СЗК.

Появляются жалобы на жгучую боль и покалывание в запястье, ладони, а также пальцах, кроме мизинца. Наблюдается болезненность и онемение, ослабление мышц, обеспечивающих движение большого пальца.

Эти заболевания обычно наступают в результате непрерывной работы на неправильно организованном рабочем месте.

Механизм нарушений, происходящих в организме под влиянием электромагнитных полей, обусловлен их специфическим (нетепловым) и тепловым действием.

Специфическое воздействие ЭМП отражает биохимические изменения, происходящие в клетках и тканях. Наиболее чувствительными являются центральная и сердечно–сосудистая системы. Возможны отклонения со стороны эндокринной системы.

В начальном периоде воздействия может повышаться возбудимость нервной системы, проявляющаяся раздражительностью, нарушением сна, эмоциональной неустойчивостью. В последующем развиваются астенические состояния, т.е. физическая и нервно–психическая слабость. Поэтому для хронического воздействия ЭМП характерны: головная боль, утомляемость, ухудшение самочувствия, гипотония (снижение артериального давления), брадикардия (урежение пульса), боли в сердце. Указанные симптомы могут быть выражены в разной степени.

Тепловое воздействие ЭМП характеризуется повышением температуры тела, локальным избирательным нагревом клеток, тканей и органов вследствие перехода ЭМП 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.

6.2 Расчетная часть

Цель работы – проектирование защиты базы данных Microsoft SQL Server на примере Microsoft Azure. Это решение позволит создать гибкую многофункциональную среду для размещения данных в удобной для пользователя среде.

На рисунке 6.1 показано расположение рабочих мест и техники.

При работе используется персональный компьютер следующей конфигурации:

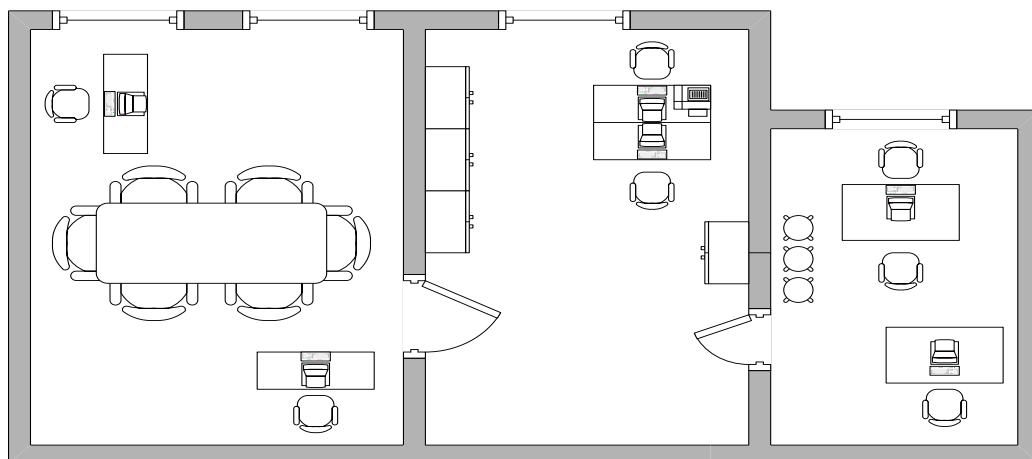


Рисунок 6.1 – План рабочего места

Материнская плата: MSI K8N Neo3, S754; Процессор: Intel Core 2 Duo E7500 (2, 93 ГГц); Оперативная память: 2Гб; Жесткий диск: MAXTOR 256GB 6T450No DiamondMax 10; Видеокарта: nVidia GeForce 9200; FDD: 3,5” Mitsumi; DVD-ROM: DVD+RW Drive LG HL-DT-ST DVDRAM GSA-H10A; Монитор: TFT 19" Samsung 971P 250 кд/м² 1500:1, 178 /178 PVA 6 мс. Мышь: ML-X200W; Клавиатура: PowerFlagshipKeyboard; Принтер: CanonImam iP1000.

6.2.1 Расчет уровня шума от персонального компьютера

Основным источником шума является системный блок ПК. Шум издают такие механические устройства, как жесткий диск, CD/DVD приводы, а так же кулеры (вентиляторы). Принтер и сканер тоже относятся к источникам шума, но ввиду того что они используются крайне редко, рассматриваться не будут.

Жесткий диск. В системе используется один жесткий диск, со скоростью вращения шпинделя 7200 об/мин. Согласно паспорту максимальный уровень шума не превышает 25 дБ при бездействии, и 32 дБ во время дисковой активности (чтение/запись/поиск).

CD/DVD приводы. В системе установлено два привода – DVD-RW. Максимальный шум из него при максимальных оборотах вращения шпинделя не должен превышать 40–45 дБ, к счастью в нынешние времена пользоваться этими приводами приходится крайне редко. Установленный FDD привод не используется и отключен.

Системы активного охлаждения. Для обеспечения циркуляции воздуха внутри системного блока и качественного охлаждения компонентов системы, в корпусе насчитывается 6 кулеров разных размеров и скоростей вращения, более подробное описание ниже.

- Кулер на процессоре – 1шт., Ø 60мм, 2700об/мин, 22дБА.
- Кулер на северном мосту материнской платы – 1шт., Ø 50мм, 4500об/мин, 23дБА.

- Кулер видеокарты – 1шт., Ø 40мм, скорость вращения и уровень шума неизвестны, но даже при выключенном компьютере и других источниках шума, шум этого кулера едва слышим.

- Кулер блока питания – 1шт., Ø 80мм, 4000об/мин, 33дБА.

- Корпусные кулеры – 2шт., 80мм, 2500об/мин, 28дБА.

Общий шум от нескольких источников не соответствует сумме шумов от каждого источника в отдельности.

Для двух находящихся рядом источников шума суммарный шум определяется следующим образом:

а) Если значения уровней шума одинаковы, то суммарный уровень шума на 3 дБ превышает уровень шума каждого источника.

б) Если разница уровней шума превышает 10 дБ, суммарный уровень шума равен величине большего из двух шумов. Например, общий шум от двух источников с уровнями 30 и 60 дБ, равен 60 дБ.

с) Если разница уровней шума не более 10 дБ, нужно воспользоваться приведенной ниже таблицей. Вычисляем разность уровней шума установок. Например, $L_1 = 52$ дБ, а $L_2 = 48$ дБ. Разность равна 4 дБ. Если источников шума более двух, метод расчета не изменяется, и источники рассматриваются парами, начиная с самых тихих.

Например, есть четыре источника шума с уровнями 25 дБ, 38 дБ, 43 дБ и 50 дБ. Сначала делаем подсчет для двух слабейших источников: $38 - 25 = 13$ дБ. Разница больше 10 дБ, и этот источник вообще не учитываем.

Для источников 38 и 43 дБ: $43 - 38 = 5$ дБ, поправка из таблицы равна 1,2 дБ. Суммарный шум трех источников: $43 + 1,2 = 44,2$ дБ.

Теперь найдем полный шум от всех источников. $50 - 44,2 = 5,8$ дБ. Округляя разность уровней шума до 6 дБ, по таблице находим поправку 1.0 дБ.

Итак, общий уровень шума от четырех источников равен $50 + 1 = 51$ дБ.

Итак, на основании вышеприведенных данных подсчитаем максимальный постоянный шум системного блока, т.е. максимальный долговременный шум. Устройства, использующиеся от случая к случаю не будут учтены (принтер, CD/DVD приводы).

Уровни шума, отсортированные по возрастанию (взяты максимальные значения):

22дБ, 23дБ, 28дБ, 28дБ, 32дБ, 33дБ

$22дБ + 23дБ = 23дБ + 2,6 = 25,6дБ$

$25,6дБ + 28дБ = 28дБ + 2,1дБ = 30,1дБ$

$30,1дБ + 28дБ = 30,1дБ + 2,1дБ = 32,2дБ$

$32,2дБ + 32 дБ = 32 дБ + 3 дБ = 35 дБ$

$35 дБ + 33 дБ = 35 дБ + 2,1 дБ = 37,1 дБ$

По полученным подсчетам, уровень шума не превышает максимально допустимый.

6.2.2 Анализ данных при выборе монитора

Технические характеристики используемого монитора TFT 19" Samsung 971P (взято из паспорта устройства):

| | |
|--------------------------------|-----------------------|
| Тип экрана: | 19.0", TFT |
| Видимая область: | 19\ |
| Точка: | 0,294 |
| Диагональ видимой области: | 480 мм |
| Максимальное разрешение: | 1440 x 900 |
| Количество цветопередачи: | 32 бита |
| Яркость: | 250 кд/м ² |
| Контрастность: | 1500:1 |
| Время отклика: | 6 мс |
| Углы обзора (по гор./по вер.): | 178/178 |
| Размеры внешние: | 423*433*228мм |
| Частота экрана: | 75 Гц |

Преимущества ЖК–мониторов TFT 19

Отсутствие излучений. Вообще, любой электроприбор испускает электромагнитные волны, но уровень излучения ЖК–монитора того же порядка, что у электробритвы или утюга, и даже меньше, чем у обычной силовой проводки, питающей розетки в помещениях.

ЭЛТ–мониторы обладают более высоким уровнем излучений по сравнению с другими электробытовыми приборами. Кроме того, помимо электромагнитного у них присутствует также бета–излучение. Хотя современные ЭЛТ–мониторы по этому параметру можно считать совершенно безопасными. По крайней мере, если это излучение и наносит какой–то вред организму, то он настолько мал по сравнению с вредом от других факторов, с которыми нам приходится сталкиваться в жизни, что пока его не удалось зафиксировать при клинических испытаниях.

Отсутствие мерцания. Действительно, в ЭЛТ–мониторе луч практически мгновенно засвечивает элемент изображения и бежит дальше по экрану. И глаз воспринимает изображение как целостное только благодаря своей инерционности. Считается, что для комфортной работы частота обновления экрана должна быть не ниже 85 Гц, в то время как «стандартная» частота ЭЛТ–дисплея составляет лишь 60 Гц. С другой стороны, увеличение частоты регенерации приводит к снижению четкости и «замыливанию» картинки. ЖК–дисплеи свободны от этого недостатка, каждый их пиксель светится на всей протяженности кадра. Правда, из–за конструктивных особенностей определенный вид мерцания присущ и ЖК–мониторам, но об этом – в разделе, посвященном их недостаткам.

Идеальная четкость. Увы, данное преимущество реализуется только в единственном «родном» разрешении экрана и только при работе с цифрового

входа. Для недорогих моделей, оснащенных лишь аналоговым входом (D-Sub), оно неактуально.

Худшая цветопередача. Это родовое свойство. Дело в том, что в ЭЛТ-мониторе изображение основано на свечении люминофора. Излучающий за счет квантовых эффектов элемент способен выдавать сколь угодно чистый цвет вплоть до когерентного. Цветообразование в ЖК-мониторе основано на излучении лампами подсветки белого цвета и затем «вырезании» из него нужных спектральных участков светофильтрами, которые принципиально не способны обеспечить спектрально-чистый цвет. Кроме того, подавляющее большинство современных ЖК-дисплеев (по крайней мере, все модели нижнего и среднего ценового диапазона) производится по TN-технологии. Цветопередача у них хуже, чем у ЖК-матриц, изготовленных по другим технологиям. Кстати, во многих моделях ЖК-дисплеев тень в правой нижней части стандартных «обоев» Windows XP выглядит фиолетовой, а не коричневой. На рисунке 6.2 показана цветопередача ЖК-мониторов ЭЛТ-мониторов.



Рисунок 6.2 – Не очень естественная цветопередача ЖК-монитора: тень имеет фиолетовый оттенок

Фиксированное разрешение. У ЖК-монитора имеется «родное» разрешение, являющееся оптимальным. При этом меньшие значения отображаются зачастую неудовлетворительно, а большие вообще не поддерживаются.

Меньшая глубина цвета. Подавляющее большинство матриц, используемых при производстве ЖК-мониторов, имеет глубину цвета 18 разрядов – по шесть на цвет, что соответствует 64 градациям яркости или 262 тыс. цветов. В то же время стандартной для компьютера считается глубина цвета 24 разряда – по восемь на цвет, т.е. 256 градаций яркости при 16,7 млн. цветов. Как говорится, почувствуйте разницу. Правда, производители мониторов предпринимают специальные меры, чтобы обеспечить непрерывный цветовой клин для 24-разрядной гаммы, при этом количество «эффективных» градаций яркости составляет 252, количество цветов – 16,2 млн. Но, с одной стороны, это достигается за счет мерцания отдельных пикселей с частотой, вдвое или вчетверо меньшей частоты регенерации экрана, т.е. 15–30 Гц. При этом доля мерцающих пикселей составляет около 75% от их общего числа. С

другой стороны, если производители панелей ограничивают количество ступеней яркости числом 64, значит, точность, которую они могут обеспечить, составляет 1–1,5%. На рисунках 6.3 а, б показаны глубины цвета мониторов 24 разряда и 32 разряда.

Инерционность. В этом направлении в последнее время достигнуты определенные успехи, что служит сейчас основным маркетинговым козырем при рекламе новых моделей. Жидкие кристаллы – огромные органические молекулы, способные вращать плоскость поляризации проходящего через них света в зависимости от ориентации. А последней можно управлять с помощью электрического поля, что и позволяет модулировать количество проходящего через них света электрическим сигналом.

Малые углы обзора. Судя по паспортным данным, анонсированные углы обзора малыми никак не назовешь, но это скорее результат особой методики измерения, чем реальные цифры. Заявленные значения контрастности дисплеев обычно составляют порядка 500, в то время как предельный угол определяется при уменьшении этого параметра до десяти. Иными словами, в число «допустимых» попадают углы обзора, при которых контрастность ухудшается в 50 раз!

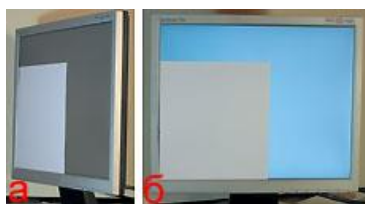


Рисунок 6.3 а, б – На обеих фотографиях лист бумаги белый. Экран дисплея – якобы тоже

Искажение цветопередачи при отклонении угла обзора от нормали. При отклонении угла обзора от центра в наиболее распространенных TFT–матрицах изображение обычно уходит в желтую или бурую области.

ЖК–монитор испускает поляризованный свет. Точные данные о его вредности для здоровья отсутствуют, но подозрение в этом остается. Так что, раз уж мы упомянули об отсутствии излучений, характерных для ЭЛТ–мониторов, в качестве преимущества ЖК, то логично будет сказать о поляризации как о недостатке. На рисунке 6.4 показано поляризация света ЭЛТ–мониторов.

При однотонном изображении верхняя и нижняя часть экрана не должны сильно отличаться по яркости. Особенно этот эффект заметен при взгляде на дисплей слегка снизу.



Рисунок 6.4 – С «точки зрения» компьютера, экран залит однородным белым цветом

Углы обзора принято измерять в вертикальной и горизонтальной плоскости, поскольку именно в таких случаях эти углы оказываются наибольшими. Наихудший же вариант – взгляд «по диагонали», например сверху слева. Используя белый лист бумаги, можно оценить также искажения цвета.

Для оценки параметров мониторов существует немало программ–тестов. Однако, так как характерные недостатки ЖК – и ЭЛТ–мониторов различны, то и методики их тестирования будут существенно различаться. Поэтому предпочтение следует отдавать программам, либо специально написанным для ЖК–мониторов, либо имеющим специфические наборы тестов.

Установлено «родное» для монитора разрешение экрана и проведено автокалибровка при работе с аналогового входа, которую можно выполнять либо на специальном калибровочном изображении, либо на картинке для проверки муара. Собственно, после автокалибровки и следует проверить монитор на наличие муара. В идеале его не должно быть.

Последовательно переберено все «чистые» цвета, которых, включая черный и белый, насчитывается восемь. При этой проверке отсутствовали «битые» пиксели, а также равномерность подсветки и чистота цвета.

7 Техничко–экономическое обоснование

7.1 Расчет затрат на разработку информационных технологий

Под информационными технологиями понимаются экономические информационные системы (ЭИС), программные продукты (ПП), информационные базы данных и т.д.

Расчет полных затрат на разработку проектного решения в виде информационных технологий ($C_{\text{пi}}$) осуществляется по формуле:

$$C_{\text{пi}} = Z_{\text{фот}} + Z_{\text{сзи}} + M_i + A + P_{\text{ми}} + П_{\text{зи}} + P_{\text{ни}}, \quad (7.1)$$

где $Z_{\text{фот}}$ – общий фонд оплаты труда разработчиков, тенге;

$Z_{\text{сзи}}$ – отчисления по социальному налогу, тенге;

M_i – затраты на материалы, тенге;

A –амортизация

$P_{\text{ми}}$ – затраты, связанные с эксплуатацией техники, тенге;

$П_{\text{зи}}$ – прочие затраты, тенге;

$P_{\text{ни}}$ – накладные расходы, тенге.

Размер фонда оплаты труда разработчиков ($Z_{\text{фот}}$) рассчитывается по формуле:

$$Z_{\text{фот}} = Z_{\text{oi}} + Z_{\text{ди}}, \quad (7.2)$$

где Z_{oi} – основная заработная плата, тенге;

$Z_{\text{ди}}$ – дополнительная заработная плата, тенге.

Затраты на оплату труда зависят от объема и трудоемкости разработки программного обеспечения[4].

Общий объем(V_0) программного продукта определяется исходя из количества и объема функции, реализуемых программой:

$$V_0 = \sum_{j=1}^n V_j, \quad (7.3)$$

где V_j – объем отдельной функции ПО;

n – общее число функций.

Расчет уточненного объема ПО представлен в таблице 7.1.

$$V_0 \approx 10000$$

Таблица 7.1. Перечень и объем функций программного модуля

| № функции | Наименование (содержание) | Объем функции (LOC) | |
|--------------|--|-------------------------------|-------------------------------|
| | | по каталогу V _i | уточненный V _{yi} |
| 1 | 2 | 3 | 4 |
| 201 | Генерация структуры базы данных | 4300 | 1000 |
| 203 | Формирование баз данных | 2180 | 800 |
| 204 | Обработка наборов и записей баз данных | 2670 | 565 |
| 205 | Обслуживание баз данных в пакетном режиме | 1260 | 622 |
| 206 | Обслуживание баз данных в интерактивном режиме | 6950 | 571 |
| 207 | Манипулирование данными | 9550 | 3890 |
| 208 | Организация поиска и поиск в базе данных | 5480 | 1260 |
| 707 | Графический вывод результатов | 480 | 1292 |
| | Итого: | 32870 | 10000 |

Общая трудоемкость небольших проектов рассчитывается по формуле:

$$T_0 = T_n \cdot K_c \cdot K_m \cdot K_n, \quad (7.4)$$

где T_n – нормативная трудоемкость;

K_c – коэффициент, учитывающий сложность ПО;

K_t – поправочный коэффициент, учитывающий степень использования при разработке стандартных модулей;

K_n – коэффициент, учитывающий степень новизны ПО.

Посредством коэффициента сложности, показанный в таблице 7.2, учитывается затраты труда, связанные со сложностью ПП [4].

В разрабатываемом дипломном проекте K_i , за счет наличия у программного модуля одновременно трех характеристик:

- режим работы в реальном времени;
- управление удаленными объектами;
- существенное распараллеливание вычислений.

Принимаем $K_i = 0,18$.

Т а б л и ц а 7.2 – Дополнительные коэффициенты сложности ПО

| Характеристика ПО | Значения K_c |
|--|----------------|
| 1. Функционирование ПО в расширенной операционной среде (связь с другими ПО) | 0,08 |

| Характеристика ПО | Значения K_c |
|--|----------------|
| 2. Интерактивный доступ | 0,06 |
| 3. Обеспечение хранения, ведения и поиска данных в сложных структурах | 0,07 |
| 4. Наличие у ПО одновременно нескольких характеристик по табл.Г4.1, приложение Г | 0,12 |
| 4.1 2 характеристики | 0,18 |
| 4.2 3 характеристики | 0,26 |
| 4.3 Свыше 3-х характеристик | |

Коэффициент, учитывающий степень использования при разработке ПО стандартных модулей (K_m). Степень использования в разрабатываемом ПО стандартных модулей определяется их удельным весом в общем объеме проектируемого продукта. В данном дипломном проекте степень охвата реализуемых функций разрабатываемого ПО стандартными модулями, типовыми программами и ПО до 20%, следовательно $K_m=0,9$.

Поправочный коэффициент, учитывающий новизну разрабатываемого ПО (K_n) определяется на основе данных представленных в таблице 7.3 и составляет 1,0

Т а б л и ц а 7.3 – Поправочные коэффициенты, учитывающие новизну ПО (K_n).

| Категория новизны | Степень новизны | Использование | | Значение K_n |
|-------------------|---|--------------------------|------------------|----------------|
| | | На основе нового типа ПК | В среде новой ОС | |
| А | Принципиально новые ПО, не имеющие доступных аналогов | + | + | 1,75 |
| | | – | + | 1,6 |
| | | + | – | 1,2 |
| | | – | – | 1,0 |
| Б | ПО, являющиеся развитием определенного параметрического ряда ПО | + | + | 1,0 |
| | | – | – | 0,9 |
| | | + | – | 0,8 |
| В | ПО, являющиеся развитием определенного параметрического ряда ПО, разработанных для ранее освоенных типов конфигурации | – | – | 0,7 |

Нормативная трудоемкость ПО (T_n) определяется на основе принятого в расчет V_y и категории сложности, которая уточняется с учетом сложности и новизны проекта и степени использования стандартных модулей при разработке.

В соответствии с этим, согласно укрупненным нормам времени на разработку ПО (T_n) в зависимости от уточненного объема ПО (V_0) и группы сложности (Приложение В) : объем ПО(строки исходного кода, LOC) 11000, категория сложности ПО 2-я – $T_n = 291$, категория сложности ПО 40.

Следовательно T_0 будет равно:

$$T_0 = 291 \cdot 0.12 \cdot 0.7 \cdot 1 = 24,44 \text{ (чел./дн.)}$$

Численность исполнителей проекта ($Ч_p$) рассчитывается по формуле:

$$Ч_p = \frac{T_0}{T_p \cdot \Phi_{эф}} \quad (7.5)$$

где $\Phi_{эф}$ – эффективный фонд времени работы одного работника в течение года (дн.);

T_0 – общая трудоемкость разработки проекта (чел./дн.);

T_p – срок разработки проекта (лет).

Срок разработки проекта (T_p) определяется по формуле:

$$T_p = \frac{T_0}{Ч_p \cdot \Phi_{эф}} \quad (7.6)$$

где $Ч_p$ – плановое число разработчиков.

Эффективный фонд времени работы одного работника ($\Phi_{эф}$) рассчитывается по формуле:

$$\Phi_{эф} = D_r - D_n - D_v - D_o, \quad (7.7)$$

где D_r – количество дней в году;

D_n – количество праздничных дней в году;

D_v – количество выходных дней в году;

D_o – количество дней отпуска.

В соответствии производственным календарем на 2014 год.: $D_r = 365$; $D_n = 14$; $D_v = 103$; $D_o = 10$, то по формуле (6.7) получим:

$$\Phi_{эф} = 365 - 14 - 103 - 10 = 228 \text{ дней}$$

Плановое число разработчиков $Ч_p = 1$, следовательно по формуле (7.6):

$$T_p = 24,44 / (1 * 228) = 0,11 \text{ года} = 40 \text{ дней}$$

Таким образом, согласно произведенным расчетам и в соответствии с формулой (7.5):

$$Ч = 24,44 / (0,11 * 228) = 1 \text{ чел.}$$

Основная заработная плата исполнителей на конкретное ПО рассчитывается по формуле:

$$Z_{oi} = \sum_{i=1}^n T_{чи} \times T_ч \times \Phi_{п} \times K \quad (7.8)$$

где n – количество исполнителей, занятых разработкой конкретного ПО;

$T_{чи}$ – часовая тарифная ставка i -го исполнителя (тыс.тенге);

$\Phi_{п}$ – плановый фонд рабочего времени i -го исполнителя (дней), 84дня;

$T_ч$ – количество часов работы в день (час), 8 часов;

K – коэффициент премирования, составляет 1,2.

По данным о специфике и сложности выполняемых функций составляется штатное расписание группы специалистов–исполнителей, участвующих в разработке ПО, с определением образования специальности, квалификации и должности (таблица 7.6).

Т а б л и ц а 7.6 – Сведения по работникам, задействованным в проекте

| Специалист – Исполнитель | Количество, человек | Зарботная плата в месяц, тенге |
|--------------------------|---------------------|--------------------------------|
| Программист | 1 | 170 000 |
| Итого | | 170 000 |

Часовая тарифная ставка рассчитывается путем деления месячной тарифной ставки на установленную при 40–часовой недельной норме рабочего времени расчетную среднемесячную норму рабочего времени в часах (Φ_p):

$$T_ч = \frac{T_m}{\Phi_p}, \quad (7.9)$$

где $T_ч$ – часовая тарифная ставка (тыс.тенге);

T_m – месячная тарифная ставка (тыс.тенге).

Таким образом:

$$T_{\text{ч}} = \frac{170\,000}{168} = 1011,4 \text{ тенге/час}$$

По формуле (7.8) можно рассчитать основную заработную плату исполнителей:

$$З_{oi} = 1011,4 * 8 * 21 * 1,38 = 234\,482 \text{ тенге}$$

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

$$З_{di} = З_{oi} * H_d / 100, \quad (7.10)$$

где H_d – коэффициент дополнительной заработной платы разработчиков 22%.

$$З_{di} = 234\,482 * 0,24 = 56\,275,68 \text{ тенге}$$

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

$$З_{czi} = (\text{ФОТ} - \text{ПО}) * 11\% \quad (7.11)$$

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

$$\text{ПО} = \text{ФОТ} * 10\% \quad (7.12)$$

Таким образом:

$$\text{ПО} = 234\,482 * 0,1 = 23\,448,2 \text{ тенге}$$

$$З_{czi} = (234\,482 - 23\,448,2) * 0,11 = 23\,213,72 \text{ тенге}$$

Затрат на материалы определяются по формуле:

$$M_i = (З_{ocн.} * H_{mз}) / 100\% \quad (7.13)$$

где $H_{mз}$ – норма расхода материалов от основной заработной платы (3–5%).

$$M_i = 234\,482 * 0,042 = 9\,848,2 \text{ тенге}$$

Амортизационные отчисления производятся по установленным нормам амортизации, выражаются, в процентах к балансовой стоимости оборудования и рассчитываются по формуле

$$A = \frac{C_{\text{обор.}} \cdot N_A \cdot N}{365 \cdot 100} \quad (7.14)$$

где N_A – норма амортизации (25 %);

$C_{\text{обор.}}$ – первоначальная стоимость оборудования;

N – фактический срок эксплуатации оборудования, 60 дней;

Данные по стоимости оборудования представлены в таблице 7.7.

Таблица 7.7 – Стоимость оборудования одного ПК с периферией

| Наименование | Цена с НДС, тенге | Цена без НДС, тенге |
|--------------------|-------------------|---------------------|
| Монитор | 32 000 | 28 160 |
| Материнская плата | 13 000 | 11 264 |
| Процессор | 30 000 | 26 400 |
| Видеокарта | 23 000 | 20 240 |
| HDD | 12 000 | 10 560 |
| DVD–RW | 6 500 | 5 720 |
| CPU Cooler | 5 500 | 4 840 |
| Оперативная память | 8 000 | 7 040 |
| Клавиатура | 1 000 | 880 |
| Мышь | 1 000 | 880 |
| Принтер | 8 000 | 7 040 |
| Итого: | 140000 | 123024 |

$$C_{\text{обор.}} = 140\,000 \text{ тенге}$$

Тогда, согласно формуле (7.14)

$$A = \frac{140\,000 \cdot 25 \cdot 60}{365 \cdot 100} = 5753,4$$

Расходы по статье «Машинное время» (P_{mi}) включают оплату машинного времени, необходимого для разработки и отладки ПО, которое

определяется по нормативам (в машино–часах) на 100 строк исходного кода (H_{MB}) машинного времени в зависимости от характера решаемых задач и типа ПК

$$P_{Mi} = C_{Mi} \times (V_{oi}/100) \times H_{MB}, \quad (7.15)$$

где C_{Mi} – цена одного машино–часа (тыс. тенге);

V_{oi} – общий объем ПО (строк исходного кода);

H_{MB} – норматив расхода машинного времени на отладку 100 строк исходного кода (машино–часов).

Согласно, учитывая данные из Приложения Д: наименование подсистемы АС и ДОС – 1 и средний расход машинного времени, ч/100 сток кода составляет 12.

Таким образом:

$$P_{Mi} = 1\,014,2 * (11\,000 / 100) * 11 = 1\,335\,048 \text{ тенге}$$

Расходы по статье «Прочие затраты» (P_{zi}) на конкретное ПО включают затраты на приобретение и подготовку специальной научно–технической информации и специальной литературы [4]. Определяются по нормативу, разрабатываемому в целом по организации, в процентах к основной заработной плате:

$$P_{zi} = Z_{oi} \times H_{pz}/100, \quad (7.16)$$

где H_{pz} – норматив прочих затрат в целом по организации в (%), в дипломной работе нужно брать 30% .

Подставляем все данные в формулу получаем:

$$P_{zi} = 234\,482 * 0,2 = 70\,344,6 \text{ тенге}$$

Затраты по статье «Накладные расходы» (P_{ni}), связанные с необходимостью содержания аппарата управления, вспомогательных хозяйств и опытных (экспериментальных) производств, а также с расходами на общехозяйственные нужды (P_{ni}), относятся на конкретное ПО по нормативу (H_{pn}) в процентном отношении к основной заработной плате исполнителей. Норматив устанавливается в целом по организации:

$$P_{ni} = Z_{oi} \times H_{pn}/100\%, \quad (7.17)$$

где P_{ni} – накладные расходы на конкретную ПО (тыс.тенге);

H_{pn} – норматив накладных расходов в целом по организации в (%), в дипломной работе нужно брать 70% .

Подставляем все данные в формулу получаем:

$$P_{ni} = 234\,482 * 0,7 = 164\,137,4 \text{ тенге}$$

$$C_{ni} = 1\,843\,061,8 \text{ тенге}$$

Результаты выполненных расчетов сводятся представлены в виде таблицу 7.8 и показаны на рисунке 7.1.

Т а б л и ц а 7.8 – Затраты на разработку

| Затраты на разработку | Условное обозначение | Значение, тенге | В процентах от общей суммы |
|-----------------------|----------------------|-----------------|----------------------------|
| Фонд оплаты труда | $Z_{\text{ФОТ}}$ | 234 482 | 12,7 |
| Социальный налог | $Z_{\text{сзп}}$ | 23 448,2 | 1,27 |
| Материалы | M_i | 9 848,2 | 0,53 |
| Амортизация | A | 5753,4 | 0,31 |
| Машинное время | $P_{\text{ми}}$ | 1 335 048 | 72,44 |
| Прочие затраты | $P_{\text{зи}}$ | 70 344,6 | 3,82 |
| Накладные расходы | $P_{\text{ни}}$ | 164 137,4 | 8,91 |
| ого: | | 1 843 061,8 | 100,00 |

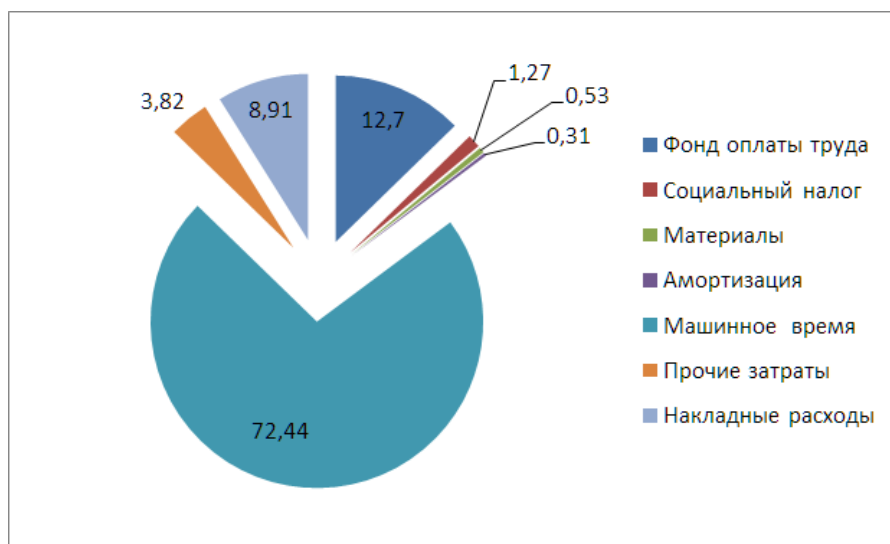


Рисунок 7.1 – Процентные доли затрат на разработку

7.2 Расчет цены программного продукта

Расчет цены ПП в организационно–экономической части дипломной работы предлагается производить следующим образом:

- если ПП разработан одной организацией по заказу другой и не предназначен для тиражирования, то затраты на разработку ПП считаются его себестоимостью, и при формировании цены применяется затратный метод;
- если ПП предназначен для тиражирования, то конечная цена определяется путем экспертных оценок на основании ценностного подхода с учетом текущих цен конкурентов (если существуют аналогичные ПП).

Расчет цены ПП, который разработан одной организацией по заказу другой и не предназначен для тиражирования, осуществляется по формуле:

$$C_{ПП} = Z_{РПР} + П_n + НДС \quad (7.18)$$

где $C_{ПП}$ – цена программного продукта, тенге;

$Z_{РПР}$ – затраты на разработку проектного решения, в данном случае программного продукта, тенге;

$П_n$ – планируемая прибыль, тенге;

$НДС$ – налог на добавленную стоимость, тенге.

Планируемая прибыль рассчитывается по формуле:

$$П_n = Z_{РПР} + R_{НПП} \quad (7.19)$$

где $R_{НПП}$ – нормативная рентабельность ПП, определяемая организацией.

$НДС$, начисленный на ПП, определяется следующим образом:

$$НДС = (Z_{РПР} + П_n) * k_{НДС} \quad (7.20).$$

где $k_{НДС}$ – ставка налога на добавленную стоимость.

Подставляем все значения в формулы (7.18) – (7.20) и получаем:

По формуле (6.28) учитывая, что $Z_{РПР} = C_{пi}$, $R_{НПП}$ – это процент рентабельности по отношению к себестоимости составляет 20% :

$$R_{НПП} = C_{пi} * 0,20 = 1\,843\,061,8 * 0,20 = 368\,612,36 \text{ тенге}$$

$$П_n = 1\,843\,061,8 + 368\,612,36 = 2\,211\,674,16 \text{ тенге}$$

Подставив данные в формуле (7.20) получаем:

$$НДС = (1\,843\,061,8 + 2\,211\,674,16) * 0,12 = 486\,568,32 \text{ тенге}$$

Подставив данные в формуле (7.18) получаем:

$$C_{\text{ПП}} = 1\,843\,061,8 + 2\,211\,674,16 + 486\,568,32 = 4\,541\,304,28 \text{ тенге}$$

7.3 Вывод

Стоимость разработки составила 1 843 061,8 тенге.

Наибольшую долю составляет машинное время.

Рыночная стоимость данного проекта (учитывая существующие аналоги) составляет 10 000 000 тенге. Таким образом, разработка и внедрение данного проекта выгодно не только с технической стороны, но и с финансовой.

Данная работа не преследует прибыли, главная цель автоматизация выявления рейтинга преподавателей.

Заключение

В данной дипломной работе была проанализирована защита Microsoft Windows Azure на примере рейтинга преподавателей. При проектировании работы были учтены все основные функции данной базы данных. Данная база данных была создана на Microsoft SQL Azure, так это наилучшая по функциональным возможностям облачная среда проектирования. В данной базе данных учтены пользовательские права, и доступ к информации разграничен. Так, например, изменять, добавлять и удалять информацию из таблиц может только администратор. А преподаватель может только просматривать информацию.

База данных содержит множество триггеров и хранимых процедур, что позволяет автоматизировать процесс обработки информации, а так же облегчить использование программы на большом количестве компьютеров. Система предполагает поддержку программы на длительный срок, так как создана с учетом вносимых изменений.

Список литературы

- 1 Сайт <http://www.oszone.net/14366/Windows-Azure>
- 2 Сайт http://ru.wikipedia.org/wiki/SQL_Azure
- 3 Сайт http://ru.wikipedia.org/wiki/Microsoft_Azure
- 4 Методические указания к выполнению экономической части дипломных работ для студентов специальности 5В070400 – Вычислительная техника и программное обеспечение Еркешева З.Д, Боканова Г.Ш. – Алматы: АУЭС, 2014.
- 5 ГОСТ ИСО/МЭК 12119–2000. Информационная технология. Пакеты программ. Требование к качеству и тестирование. – М.: Издательство стандартов, 2000.
- 6 ГОСТ 28195–89. Оценка качества программных средств. Общие положения. – М.: Издательство стандартов, 1989.
- 7 Международный стандарт ISO/IEC 9126–1:2001. Программирование. Качество продукта. Часть 1. Модель качества.
- 8 Сборник временных норм на работы по ведению Государственного мониторинга геологической среды (состояния недр), информационной деятельности, цифровому картографированию. – Томск: ОГУП ТЦ Томского мониторинг, 2008.
- 9 Сатимова Е.Г. Проектирование баз данных: Методические указания к выполнению лабораторных работ (для студентов всех специальностей). – Алматы: АИЭС, 2009.
- 10 MICROSOFTSQLSERVER 2008. Реализация и обслуживание. Учебный курс Microsoft / Пер. с английского. – М.: «Русская редакция», Спб.: «Питер», 2007.

Приложение А

Классификация типов программного обеспечения

Таблица А.1 – Классификация типов программного обеспечения

| Код типа | Наименование типа ПО | Состав и содержание типа ПО |
|----------|---|--|
| 1.0 | ПО общего назначения | <p>1.1 ПО СУБД.</p> <p>1.2 ПО систем ведения линейных файлов.</p> <p>1.3 ПО ведения баз данных и линейных файлов.</p> <p>1.4 ПО информационно-поисковых и информационно-справочных систем.</p> <p>1.5 ПО ввода информации.</p> <p>1.6 ПО мониторов телеобработки и сетей ПЭВМ.</p> <p>1.7 ПО окружения СУБД, расширяющие возможности существующих СУБД.</p> <p>1.8 ПО, расширяющие возможности обработки.</p> |
| 2.0 | ПО технологии автоматизации программирования и проектирования АСУ | <p>2.1 ПО автоматизации проектирования для автоматизации проектирования различных АСУ.</p> <p>2.2 ПО технологии программирования.</p> <p>2.3 ПО автоматизации программирования (для автоматизации процессов обработки и вывода информации).</p> <p>2.4 ПО, расширяющие существующие языки программирования для повышения их компактности и простоты пользования.</p> <p>2.5 ПО общего назначения, функционально-ориентированные.</p> <p>2.6 ПО автоматического программирования.</p> <p>Реализуют различные классы экономико-математических методов и являются системами общего назначения, которые применяются в различных АСУ, для научно-технических расчетов и исследований.</p> |
| 3.0 | ПО методоориентированных расчетов | <p>3.1 ПО оптимизационных расчетов (обеспечивают решение различного класса задач оптимального планирования и управление производством).</p> <p>3.2 ПО статистического анализа и прогнозирования (для прогнозирования ТЭП, спроса и т.д.).</p> <p>3.3 ПО сетевого планирования и управления.</p> <p>3.4 ПО общей математики.</p> <p>3.5 ПО имитационного моделирования.</p> |
| 4.0 | ПО организации вычислительного процесса | <p>Автоматизация процесса ведения наборов данных, при обеспечении их надежного и систематизированного хранения.</p> <p>Повышение производительности ПЭВМ и пользователей ПО.</p> <p>Формирование и выдача отчетов о работе ПЭВМ.</p> <p>Оперативный контроль системы и ресурсов.</p> <p>Для автоматизации обработки экономических данных, при этом выделяются ПО, несущие функциональную нагрузку в АСУ.</p> <p>ПО данного типа выполнены в основном автономно.</p> |
| 5.0 | ПО функционального назначения | <p>5.1 ПО системы ПОП и СУП.</p> <p>5.2 ПО оперативного управления основным производством.</p> <p>5.3 ПО управления технической подготовкой производства.</p> <p>5.4 ПО бухгалтерского учета и управления финансами.</p> <p>5.5 ПО управления кадрами.</p> |

Приложение Б

Каталог функций программного обеспечения

Таблица Б.1 – Каталог функций программного обеспечения

| № п/п | Наименование (содержание) функций | Объем функций (строк исходного кода) | | |
|---|---|--|-----------------------|------------------------|
| | | С использованием среды разработки приложений | | |
| | | Delphi (Borland) | C++ Builder (Borland) | Visual C++ (Microsoft) |
| 1. Ввод, анализ входной информации, генерация кодов и процессор входного языка | | | | |
| 101 | Организация ввода информации | 100 | 110 | 150 |
| 102 | Контроль, предварительная обработка и ввод информации | 550 | | |
| 103 | Анализ входного языка (синтаксический и семантический) | 630 | 660 | 980 |
| 104 | Преобразование операторов входного языка и команды другого языка | 1050 | 1050 | 980 |
| 105 | Обработка входного заказа и формирование таблиц | 750 | 900 | 1340 |
| 106 | Преобразование входного языка в машинные команды (транслятор, препроцессор, макрогенератор) | 1300 | 4300 | 5100 |
| 107 | Синтаксический и семантический анализ входного языка и генерация кодов команд | 8700 | 5400 | 5400 |
| 108 | Процессор языка | 3000 | 2300 | 2300 |
| 109 | Организация ввода/вывода информации в интерактивном режиме | 220 | 220 | 320 |
| 110 | Организация ввода/вывода информации с сети терминалов | 3680 | 3340 | 3200 |
| 110 | Управление вводом/выводом | 2700 | 2900 | 2400 |
| 2. Формирование, введение и обслуживание баз данных | | | | |
| 201 | Генерация структуры базы данных | 3450 | 3950 | 4300 |
| 202 | Генерация структуры базы данных | 1540 | 1610 | 2060 |
| 203 | Формирование баз данных | 1700 | 1750 | 2180 |
| 204 | Обработка наборов и записей базы данных | 2050 | 2350 | 2670 |
| 205 | Обслуживание базы данных в пакетном режиме | 1030 | 1100 | 1260 |
| 206 | Обслуживание базы данных в интерактивном режиме | 3800 | 4400 | 6950 |
| 207 | Манипулирование данными | 8400 | 8670 | 9550 |
| 208 | Организация поиска и поиск в базе данных | 5230 | 5460 | 5480 |
| 209 | Реорганизация базы данных | 130 | 190 | 220 |
| 210 | Загрузки базы данных | 3150 | 2950 | 2780 |
| 3. Формирование и обработка файлов | | | | |
| 301 | Формирование последовательного файла | 340 | 360 | 290 |
| 302 | Сортировка файла | 340 | 360 | 290 |

Продолжение приложения Б

Окончание таблицы Б.1

| | | | | |
|--|--|-------|-------|-------|
| 304 | Обслуживание файлов | 520 | 540 | 420 |
| 305 | Обработка файлов | 750 | 800 | 720 |
| 306 | Обработка файлов в диалоговом режиме | 2400 | 2600 | 3050 |
| 307 | Совместная обработка группы файлов | 4900 | 5300 | 6180 |
| 308 | Управление файлами | 5130 | 5380 | 5750 |
| 309 | Формирование файла | 1100 | 1080 | 1020 |
| 4. Генерация программ и ПО, а также настройка ПО | | | | |
| 401 | Генерация рабочих программ | 3680 | 4120 | 3360 |
| 402 | Генерация программ по описанию пользователей | 10870 | 12330 | 9880 |
| 403 | Формирование служебных таблиц | 570 | 620 | 1070 |
| 404 | Система генерации ПО | 5120 | 5340 | 4980 |
| 405 | Система генерации ПО | 250 | 300 | 370 |
| 5. Управление ПО, компонентами ПО и внешними устройствами | | | | |
| 501 | Монитор ПО (управление работой компонентов) | 350 | 360 | 740 |
| 502 | Монитор системы (управление работой комплекса ПО) | 3750 | 3880 | 7740 |
| 503 | Управление внешними устройствами и объектами | 6850 | 7340 | 5900 |
| 504 | Обработка прерываний | 890 | 730 | 540 |
| 505 | Управление внешней памятью | 250 | 210 | 200 |
| 506 | Обработка ошибочных и сбойных ситуаций | 430 | 410 | 410 |
| 507 | Обеспечение интерфейса между компонентами | 730 | 750 | 970 |
| 6. Отладка прикладных программ, обмен информацией между магнитным диском и магнитной лентой, вспомогательные программные функции | | | | |
| 601 | Отладка прикладных программ в интерактивном режиме | 4500 | 4700 | 4300 |
| 602 | Обмен информацией между магнитным диском и магнитной лентой | * | * | * |
| 603 | Копирование наборов данных на магнитной ленте и восстановление | * | * | * |
| 604 | Справка и обучение | 680 | 680 | 720 |
| 605 | Вспомогательные и сервисные программы | 460 | 490 | 580 |
| 7. Расчетные задачи, формирование и вывод на внешние носители (АЦПУ) документов сложной формы и файлов | | | | |
| 701 | Матстатистика и прогнозирование | 8370 | 9570 | 9320 |
| 702 | Расчетные задачи (расчет режимов обр-ки) | 12600 | 15300 | 14800 |
| 703 | Расчет показателей | 410 | 500 | 460 |
| 704 | Процессор отчетов | 1070 | 1230 | 3200 |
| 705 | Формирование и вывод на внеш.носители | 2650 | 2850 | 3500 |

Приложение В
Укрупненные нормы времени на разработку ПО в зависимости от
уточненного объема ПО и группы сложности ПО

Таблица В.1 – Укрупненные нормы времени на разработку ПО (T_H) в зависимости от уточненного объема ПО (V_y) и группы сложности ПО (чел./дн.)

| Объем ПО(строки исходного кода, LOC) | Категории сложности ПО | | | Категории сложности ПО |
|--------------------------------------|------------------------|-----|-----|------------------------|
| | 1-я | 2-я | 3-я | |
| <i>I</i> | 2 | 3 | 4 | 5 |
| 200 | – | – | 21 | 1 |
| 300 | – | – | 23 | 2 |
| 400 | – | – | 25 | 3 |
| 500 | – | – | 27 | 4 |
| 600 | – | 33 | 28 | 5 |
| 700 | – | 36 | 30 | 6 |
| 800 | – | 38 | 32 | 7 |
| 900 | – | 40 | 34 | 8 |
| 1000 | 51 | 43 | 36 | 9 |
| 1200 | 54 | 45 | 38 | 10 |
| 1400 | 57 | 48 | 40 | 11 |
| 1600 | 60 | 50 | 42 | 12 |
| 1800 | 64 | 54 | 45 | 13 |
| 2000 | 68 | 57 | 48 | 14 |
| 2200 | 73 | 61 | 51 | 15 |
| 2400 | 76 | 64 | 54 | 16 |
| 2600 | 81 | 68 | 57 | 17 |
| 2800 | 86 | 72 | 60 | 18 |
| 3000 | 91 | 76 | 64 | 19 |
| 3200 | 97 | 81 | 68 | 20 |
| 3400 | 103 | 86 | 72 | 21 |
| 3600 | 110 | 92 | 77 | 22 |
| 3800 | 117 | 98 | 82 | 23 |
| 4000 | 124 | 104 | 87 | 24 |
| 4200 | 133 | 111 | 93 | 25 |
| 4400 | 141 | 118 | 99 | 26 |
| 4600 | 151 | 126 | 105 | 27 |
| 4800 | 160 | 134 | 112 | 28 |
| 5000 | 170 | 142 | 119 | 29 |
| 5500 | 182 | 152 | 127 | 30 |
| 6000 | 194 | 162 | 135 | 31 |
| 6500 | 206 | 172 | 144 | 32 |
| 7000 | 220 | 184 | 154 | 33 |
| 7500 | 235 | 196 | 164 | 34 |
| 8000 | 252 | 210 | 175 | 35 |
| 8500 | 268 | 224 | 187 | 36 |
| 9000 | 288 | 240 | 200 | 37 |
| 9500 | 307 | 256 | 214 | 38 |
| 10000 | 327 | 273 | 228 | 39 |

Продолжение приложения В

Окончание таблицы В.1

| Объем ПО(строки исходного кода, LOC) | Категории сложности ПО | | | Категории сложности ПО |
|---|------------------------|----------|----------|---------------------------|
| | 1-я | 2-я | 3-я | |
| <i>1</i> | <i>2</i> | <i>3</i> | <i>4</i> | <i>5</i> |
| 11000 | 349 | 291 | 243 | 40 |
| 12000 | 374 | 312 | 260 | 41 |
| 13000 | 399 | 333 | 278 | 42 |
| 14000 | 427 | 356 | 297 | 43 |
| 15000 | 456 | 380 | 317 | 44 |
| 16000 | 487 | 406 | 339 | 45 |
| 18000 | 520 | 434 | 362 | 46 |
| 20000 | 556 | 464 | 387 | 47 |
| 22000 | 595 | 496 | 414 | 48 |
| 24000 | 636 | 530 | 442 | 49 |
| 26000 | 679 | 566 | 472 | 50 |
| 28000 | 727 | 606 | 505 | 51 |
| 30000 | 775 | 646 | 540 | 52 |
| 32000 | 830 | 692 | 577 | 53 |
| 34000 | 888 | 740 | 617 | 54 |
| 36000 | 950 | 792 | 660 | 55 |
| 38000 | 1016 | 847 | 706 | 56 |
| 40000 | 1087 | 906 | 755 | 57 |
| 42000 | 1161 | 968 | 807 | 58 |
| 44000 | 1242 | 1035 | 863 | 59 |
| 46000 | 1328 | 1107 | 923 | 60 |
| 48000 | 1420 | 1184 | 987 | 61 |
| 50000 | 1620 | 1267 | 1056 | 62 |

Приложение Г
Характеристики категорий сложности ПО

Таблица Г.1 – Характеристики категорий сложности ПО

| Категории сложности | Характеристики ПО |
|---------------------|---|
| 1 | ПО, обладающие одной или несколькими из следующих характеристик: 1) Наличие сложного интеллектуального языкового интерфейса с пользователем. 2) Режим работы в реальном времени. 3) Обеспечение телекоммуникационной обработки данных и управление удаленными объектами. 4) Машинная графика. 5) Многомашинные комплексы. 6) Обеспечение существенного распараллеливания вычислений |
| 2 | ПО, обладающие одной или несколькими из следующих характеристик: 1) Оптимизационные расчеты. 2) Обеспечение настройки ПО на изменения структур входных и выходных данных. 3) Настройка ПО на нестандартную конфигурацию технических средств. 4) Обеспечение переносимости ПО. 5) Реализация особо сложных инженерных и научных расчетов. |
| 3 | ПО, не обладающие перечисленными выше характеристиками. |

Приложение Д

Оценка значений среднего машинного времени на отладку 100 строк исходного кода без применения ПО

Таблица Д.1 – Оценка значений среднего машинного времени на отладку 100 строк исходного кода без применения ПО

| Наименование подсистемы АС и СОД | Средний расход машинного времени, ч/100 строк кода |
|--|--|
| 1. Общесистемные задачи: ведение линейных файлов, информационно–поисковые системы и информационно–справочные системы, сбор информации, ввод информации, расширение возможностей средств обработки данных, организация вычислительного процесса | 12 |
| 2. Задачи расчетного характера | 15 |
| 3. Оперативное управление производством, расчеты по ценообразованию | 7 |
| 4. Техническая подготовка производства, транспортное, ремонтное, энергетическое и инструментальное обслуживание производства | 8 |
| 5. Бухгалтерский учет, финансовые расчеты, учет пенсий и пособий, учет страховых операций, качество продукции | 13 |
| 6. Управление кадрами | 13 |
| 7. Техничко–экономическое планирование | 13 |
| 8. Материально–техническое снабжение, реализация и сбыт готовой продукции | 13 |