

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

кафедра Компьютерных технологий

«Допущен к защите»
Заведующий кафедрой _____

(Ф.И.О., ученая степень, звание)

« _____ » _____ 20__ г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Информационная система «Бюджетное планирование»

Специальность БВО70400 ВТ-ПО

Выполнил (а) Алиатов С.Т. ВТ-12-1
(Фамилия и инициалы) группа

Научный руководитель Кортсизбаев Т.Т. к.ф.и.н., доцент
(Фамилия и инициалы, ученая степень, звание)

Консультанты:

по экономической части:

Бекмусова А.С., к.э.н., доцент

(Фамилия и инициалы, ученая степень, звание)

А.С. « 31 » 03 2016 г.
(подпись)

по безопасности жизнедеятельности:

Приходько Н.Г., д.х.н., профессор

(Фамилия и инициалы, ученая степень, звание)

Н.Г. « 12 » 05 2016 г.
(подпись)

по применению вычислительной техники:

Кортсизбаев Т.Т. к.ф.и.н., доцент

(Фамилия и инициалы, ученая степень, звание)

Т.Т. « 1 » 06 2016 г.
(подпись)

Нормоконтролер: Кортсизбаев Т.Т. к.ф.и.н., доцент
(Фамилия и инициалы, ученая степень, звание)

Т.Т. « 1 » 06 2016 г.
(подпись)

Рецензент:

(Фамилия и инициалы, ученая степень, звание)

« _____ » _____ 20__ г.
(подпись)

Алматы 2016 г.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Факультет Аэрокосмических и информационных технологий
Специальность Высшая инженерная техника и программное обеспечение
Кафедра Компьютерных технологий

ЗАДАНИЕ

на выполнение дипломного проекта

Студент Нематов Шауэр Тулеубов
(фамилия, имя, отчество)

Тема проекта Информационная система «Бюджетное планирование»

утверждена приказом ректора № 148 от «19» октября 2015 г.

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

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

Разработка базы данных «Бюджетное планирование»
Разработать приложение для базы данных

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

Анализ предметной области
Изучение логической структуры
Подготовка базы данных
Проектирование базы данных «Бюджетное планирование»
Расчетная часть
Техно-экономическая обоснование
Безопасность и надежность

Перечень графического материала (с точным указанием обязательных чертежей)

- структура работы (схема)
- диаграмма компонентов
- диаграмма прецедентов
- диаграмма развертывания
- диаграмма интеграции

Рекомендуемая основная литература

- Роб П. Корона К. Системы базы данных: проектирование, реализация и управление
- Ильинер Д. Основы самостоятельного ИТ
- Г.М. Букаева Экономика от А до Я: Тематический справочник
- Методическое указание к выполнению экономической части дипломных работ для студентов специальности ВТ и ТИ

Консультанты по проекту с указанием относящихся к ним разделов

Раздел	Консультант	Сроки	Подпись
БЖР	Брихобасов И.Г.	14.03 - 12.05.16	[Подпись]
Эконом. часть	Букаева А.И.	17.03 - 31.05.16	[Подпись]

Аннотация

В данной дипломной работе разрабатывается информационная система «Бюджетное планирование».

Разработан интерфейс и база данных «Бюджетное планирование» с целью упрощения работы управления экономики и бюджетного планирования.

Интерфейс разработан с использованием новейших технологий отвечает всем современным требованиям к программному обеспечению.

Так же в последних главах рассматривается технико-экономическое обоснование и рассчитывается цена разработки проекта и рассматриваются вопросы безопасности жизнедеятельности.

Abstract

In this diploma project is developed information system “budget planning”.

Designed interface and database “budget planning” in order to simplify work to management of economy and budget planning.

The interface is designed using the latest technology and meets all modern requirements to software.

Also in the last chapter provide a feasibility study and calculate the price of the project, and addresses safety.

Андатпа

Осы дипломдық жұмыста «Бюджеттік жоспарлау» ақпарат жүйе жүргізіледі.

Экономика және бюджеттік жоспарлау басқармасы жеңілдету мақсатында деректер базасын интерфейс және « бюджеттік жоспарлау» деректер.

Интерфейс жаңа технологияларды пайдалана отырып, әзірленген және бағдарламалық қамтамасыз ету үшін барлық заманауи талаптарға сай келеді.

Сондай-ақ, соңғы тарауларда техникалық-экономикалық негіздемесін қамтамасыз ету және жобаның бағасын есептеу, сондай-ақ қауіпсіздікті қарастырады.

Содержание

Введение.....	16
1 Теоретическая часть	17
1.1 Постановка задачи	17
1.1.1. Основные технические решения.....	19
1.2 Этап начальной разработки базы данных	20
1.2.1 Анализ аналогичных приложений.....	20
1.2.2 Структура работы Управления экономики и бюджетного планирования.....	20
1.2.3 UML диаграммы	21
1.3 Проектирование базы данных	25
1.3.1 Определение атрибутов всех сущностей	25
1.4 Используемые технологии программирования	45
1.4.1 СУБД Sybase (ASE).....	45
1.4.2 Delphi 7, Perl.....	45
2 Расчетная часть	46
2.1 Расчет места для хранения базы данных.....	46
2.2 Меры по обеспечению безопасности.....	56
3 Разработка ПО	57
3.1 Разработка интерфейса ИС «Бюджетное планирование»	57
4 Технико-экономическое обоснование	63
4.1 Описание работы и обоснование необходимости	63
4.2 Расчет затрат на разработку информационных технологий	63
4.3 Оборудование, используемое в работе.....	68
4.4 Программное обеспечение, используемое в работе.....	68
4.5 Расчет амортизационных отчислений	69
4.6 Расчет затрат на электроэнергию.....	69
4.7 Расчет цены программного продукта	71
5 Безопасность жизнедеятельности	72
5.1 Анализ условий рабочего помещения	72
5.2 Расчет системы кондиционирования кабинета.	73
5.3 Расчет искусственного освещения точечным методом	76
Заключение	79
Список литературы	80

Введение

Ежегодно Управление экономики и бюджетного планирования формирует и представляет в Мажилис проект бюджета на следующий финансовый год и контролирует выполнение бюджета в течение текущего финансового года. В процессе формирования проекта бюджета все бюджетные ведомства г. Алматы представляют данные на бумажных носителях. Данные поступают в отделы Управление экономики и бюджетного планирования. Свод данных по формированию бюджета в целом по городу выполняет сводный отдел экономического планирования. В настоящее время весь процесс формирования проекта бюджета выполняется вручную и требует больших затрат времени и труда.

Основной целью создания информационной системы «Бюджетное планирование» является автоматизация функций сотрудников Управление для повышения оперативности при расчетах и анализе обоснования бюджетных программ на стадии их планирования на следующий финансовый год.

Система предназначена для:

- ввода и обработки данных при формировании бюджета;
- накопления базы данных Бюджетных программ г.Алматы.
- разработки проектов бюджета на среднесрочный период и очередной финансовый год;
- контроля над выполнением бюджетных проектов (программ);
- получение отчетов для проведения экономического анализа и планирования.

1 Теоретическая часть

1.1 Постановка задачи

Одной из основных задач Департамента является экономически обоснованное планирование доходов и расходов бюджета города и планомерное распределение бюджетных средств.

2.1.2.1. Бизнес процесс - прогнозирование поступлений бюджета. Процесс не автоматизированный.

Прогнозирование поступлений бюджета осуществляется отделом прогнозирования поступлений при содействии Департамента финансов на основе среднесрочного плана социально-экономического развития региона и среднесрочной фискальной политики на предстоящий трехлетний период.

Поступления в бюджет отражаются в виде стоимости товаров (работ, услуг). При этом в расходах должна предусматриваться бюджетная программа в объеме, равном стоимости полученных товаров (работ, услуг).

Прогнозные показатели поступлений бюджета на предстоящий финансовый год рассматриваются и одобряются бюджетной комиссией.

2.1.2.2. Бизнес-процесс - определение лимитов расходов бюджета. Процесс не автоматизированный.

При разработке проекта бюджета на предстоящий финансовый год Департамент формирует лимиты расходов местного бюджета для текущих бюджетных программ и программ развития.

Отдел бюджетной политики и планирования на основе среднесрочной фискальной политики распределяет лимиты расходов бюджета для текущих бюджетных программ и бюджетных программ развития.

Лимиты расходов бюджета рассматриваются и одобряются бюджетной комиссией.

Согласованные лимиты расходов местного бюджета для текущих бюджетных программ и программ развития через отделы Департамента (бюджетного планирования в сфере образования, культуры, спорта и информационного пространства, бюджетного планирования в сфере здравоохранения и социальной защиты населения, бюджетного планирования государственного аппарата и правоохранительных органов, бюджетного планирования развития реального сектора) доводятся до администраторов программ.

2.1.2.3. Бизнес-процесс - составление бюджетных заявок администраторами бюджетных программ. Процесс не автоматизированный

В соответствии с выделенными лимитами администраторы программ распределяют расходы по программам, подпрограммам и мероприятиям.

Распределенные расходы оформляются в виде бюджетных заявок в соответствии с формами, утвержденными Правилами составления и представления бюджетной заявки администраторами бюджетных программ.

Бюджетная заявка представляет собой совокупность документов, ежегодно составляемых и представляемых администратором бюджетных программ на предстоящий финансовый год и трехлетний период для обоснования необходимости бюджетных программ и определения суммы их финансирования при разработке бюджета.

Бюджетная заявка составляется на основе:

- единой бюджетной классификации;
- лимитов расходов бюджета;
- бюджетной заявки на трехлетний период, одобренной бюджетной комиссией в истекшем финансовом году;
- среднесрочного плана социально-экономического развития республики или региона и среднесрочной фискальной политики;
- перечня приоритетных бюджетных инвестиционных проектов (программ).

Бюджетная заявка администратора бюджетных программ включает в себя:

- 1) перечень бюджетных программ (подпрограмм) и суммы расходов по ним;
- 2) пояснительную записку к бюджетной программе;
- 3) проект паспорта бюджетной программы (подпрограммы), включенной в бюджетную заявку;
- 4) другую информацию.

Расчеты расходов по бюджетной программе (подпрограмме) раскрывают количественные и стоимостные показатели бюджетных программ, определяемые с учетом утвержденных натуральных норм.

В случае если администратором бюджетных программ предлагается реализация бюджетных программ посредством бюджетного кредитования, бюджетная заявка по данным бюджетным программам включает обоснование условий бюджетных кредитов, расчет окупаемости бюджетного кредитования и т.д.

1.2.2.4. Бизнес-процесс - рассмотрение бюджетных заявок. Процесс не автоматизированный.

Администратор бюджетной программы вносит бюджетную заявку в пределах лимита расходов.

Информация поступает в отделы бюджетного планирования в сфере образования, культуры, спорта и информационного пространства, бюджетного планирования в сфере здравоохранения и социальной защиты населения, бюджетного планирования государственного аппарата и правоохранительных органов, бюджетного планирования развития реального сектора экономики Департамента. После согласования с отделами с

вышеперечисленными отделами бюджетной заявки передаются в отдел бюджетной политики и планирования. Отдел бюджетной политики и планирования проводит их оценку и готовит по ним соответствующее заключение и готовит свод для бюджетной комиссии.

Бюджетная комиссия рассматривает бюджетные заявки и заключение Департамента по бюджетному планированию и принимает по ним решение.

1.2.2.5. Бизнес-процесс - ввод информации с бюджетных заявок в таблицы БД – разработан интерфейс ввода данных по доходам и затратам.

Отдел прогнозирования поступлений вводит прогнозные данные по поступлениям в бюджет согласованный с Налоговым комитетом.

Отделы бюджетного планирования в сфере образования, культуры, спорта и информационного пространства, бюджетного планирования в сфере здравоохранения и социальной защиты населения, бюджетного планирования государственного аппарата и правоохранительных органов, бюджетного планирования развития реального сектора Департамента вводят данные по расходной части проекта бюджета.

В результате формируется БД с данными по проекту бюджета на следующий финансовый год.

1.2.2.6. Бизнес-процесс – формирование документа «Проект бюджета города республиканского значения, столицы» на соответствующий финансовый год. Процесс автоматизирован.

Свод данных по формированию проекта бюджета в целом по городу выполняет отдел бюджетной политики и планирования. Уточнение и согласование расходов по всем программам, подпрограммам и мероприятиям с администраторами программ производиться в рабочем порядке.

Сформированный проект бюджета вносится на рассмотрение в бюджетную комиссию.

После рассмотрения проекта бюджета в бюджетной комиссии проект представляется для согласования в постоянную бюджетную комиссию Маслихата для рассмотрения.

Постоянная бюджетная комиссия Маслихата осуществляет свод предложений по внесению изменений и дополнений в проект местного бюджета, который выносится для обсуждения на сессию соответствующего Маслихата.

Утверждается местный бюджет на соответствующий финансовый год посредством принятия решения Маслихатом.

Утвержденный бюджет на следующий финансовый год в БД помечается как принятый и корректировка данных по суммам утвержденного бюджета блокируется.

1.1.1. Основные технические решения

2.1.3.1. Бизнес-архитектура

Логическое представление бизнес-архитектуры - для хранения информации использует один уровень хранения и обработки данных –

уровень Департамента экономики и бюджетного планирования городского Акимата.

Сервер физически должен располагаться в помещении Акимата города.

В общем случае предполагается нижеследующая схема организации функционирования при формировании проекта бюджета на следующий финансовый год.

Сотрудники Департамента через коммуникационную среду посредством программы клиента подключаются к серверу городского Акимата. На клиентской рабочей станции программное обеспечение построено по принципу «толстого клиента». На сервере располагается вся база данных и хранимые процедуры.

Каждый сотрудник Департамента (кроме отдела прогнозирования доходов), участвующий в процессе формирования бюджета вводит данные из бюджетных заявок, предоставленных администраторами бюджетных программ. По окончании работы по вводу данных информация из рабочих таблиц «Бюджет по отраслям экономики (расходы)» переносится в основную таблицу «Бюджет по расходам». При работе сотрудников отдела прогнозирования доходов, введенные данные по доходам записываются сразу в основную таблицу «Бюджет по доходам» без пересылок.

2.1.3.2. Решения по системной архитектуре

По характеру выполняемых функций, АИС "Бюджетное планирование" разделяется на три подсистемы:

- подсистема «Ввод данных» - отвечает за выполнение операций ввода и корректировки данных;

- подсистема "Отчеты" - отвечает за формирование отчетов на основе создаваемой информационной базы;

- подсистема «Администрирование» - отвечает за реализацию обеспечивающих функций: ограничение доступа к системе, администрирование пользователей Системы и НСИ, взаимодействие компонентов;

1.2 Этап начальной разработки базы данных

1.2.1 Анализ аналогичных приложений

В виду того что эта система сделана специально для Управления экономики и бюджетного планирования и до этой системы у Управления не было аналогов и они вручную создавали отчеты и хранили на своих компьютерах. Поэтому мое программное обеспечение является уникальной и специфичной для нужд Управления.

1.2.2 Структура работы Управления экономики и бюджетного планирования

Для каждого отдела Управления существует свои структуры работы. Основной задачей является создание бюджетных программ на стадии их планирования на следующий финансовый год. Для этого каждый отдел создает проект бюджета на следующий год где рассчитывает расходы для

ЕБК своего отдела. Так же они рассчитывают в каждый квартал уточнения. Ко всему прочему отдел доходов собирает фактический каждодневный доход с помощью интеграции с Казначейством и не посредственной выгрузки отчетов ежедневного поступления. Для отчетности с руководством Управления и вышестоящих органов отделы расходов и доходов создают отчеты по плановым и фактическим данным.

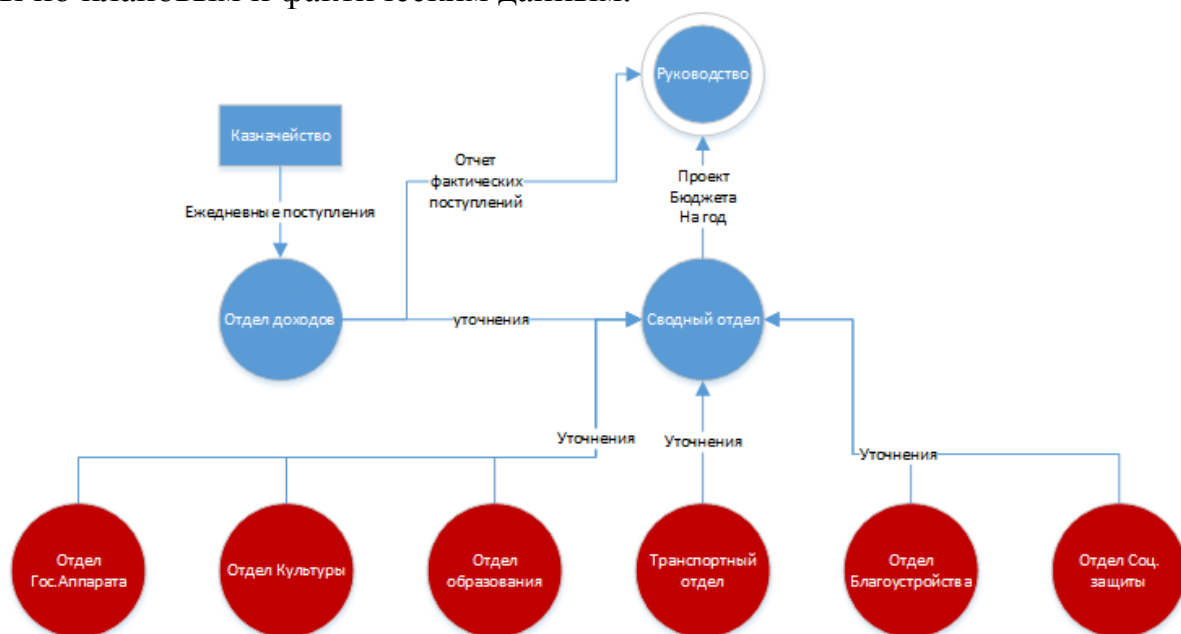


Рисунок 1.1- Структура работы

1.2.3 UML диаграммы

Для проектирования базы данных «Бюджетное планирования» необходимо основные диаграммы UML.

1. Диаграмма компонентов;
2. Диаграмма прецедентов;
3. Диаграмма развертывания.

Диаграмма компонентов

Диаграмма компонентов, Component diagram — статическая структурная диаграмма, показывает разбиение программной системы на структурные компоненты и связи (зависимости) между компонентами.[1].

В данном случае исполняемый модуль budget.exe подключается для добавления или просмотра данных к БД budget.dat с помощью посредника СУБД Sybase. Поэтому случаю СУБД Sybase является посредником между исполнительной программой и БД, обеспечивая их взаимодействие. Диаграмма компонентов на рисунке 1.2

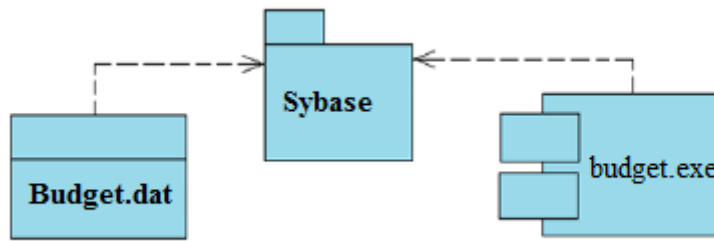


Рисунок 1.2 – Диаграмма компонентов

Диаграмма прецедентов

Для упрощения работы пользователей так же для обеспечения безопасности данных в БД объявляются группы и пользователи со своими привилегиями.

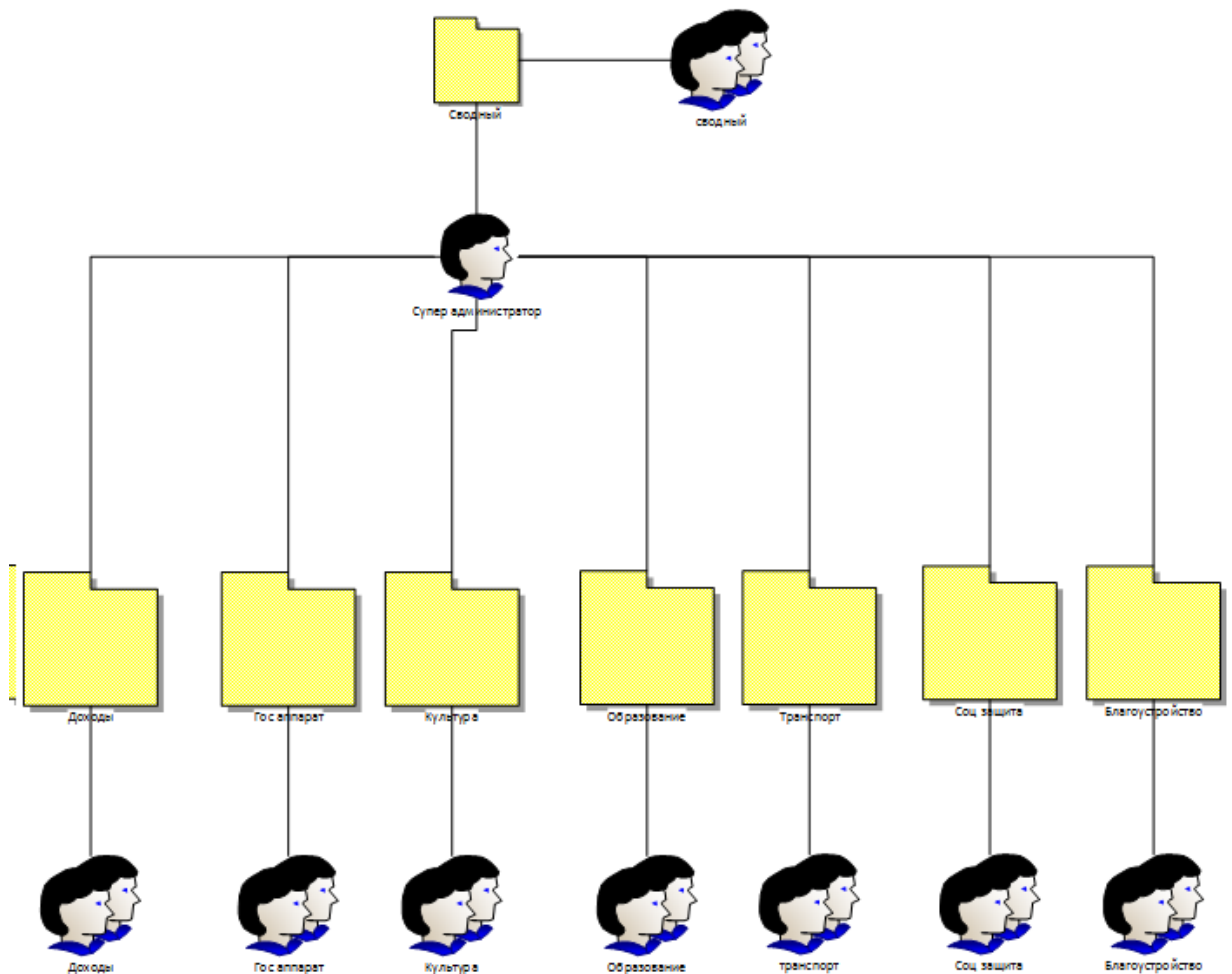


Рисунок 1.3 – Диаграмма прецедентов

На рисунке 1.3 показана диаграмма прецедентов для групп. По данной диаграмме следует что доступ к группе таблиц для каждого отдела имеют только пользователи данного отдела кроме администратора, у которого есть привилегии в доступе к любой таблице.

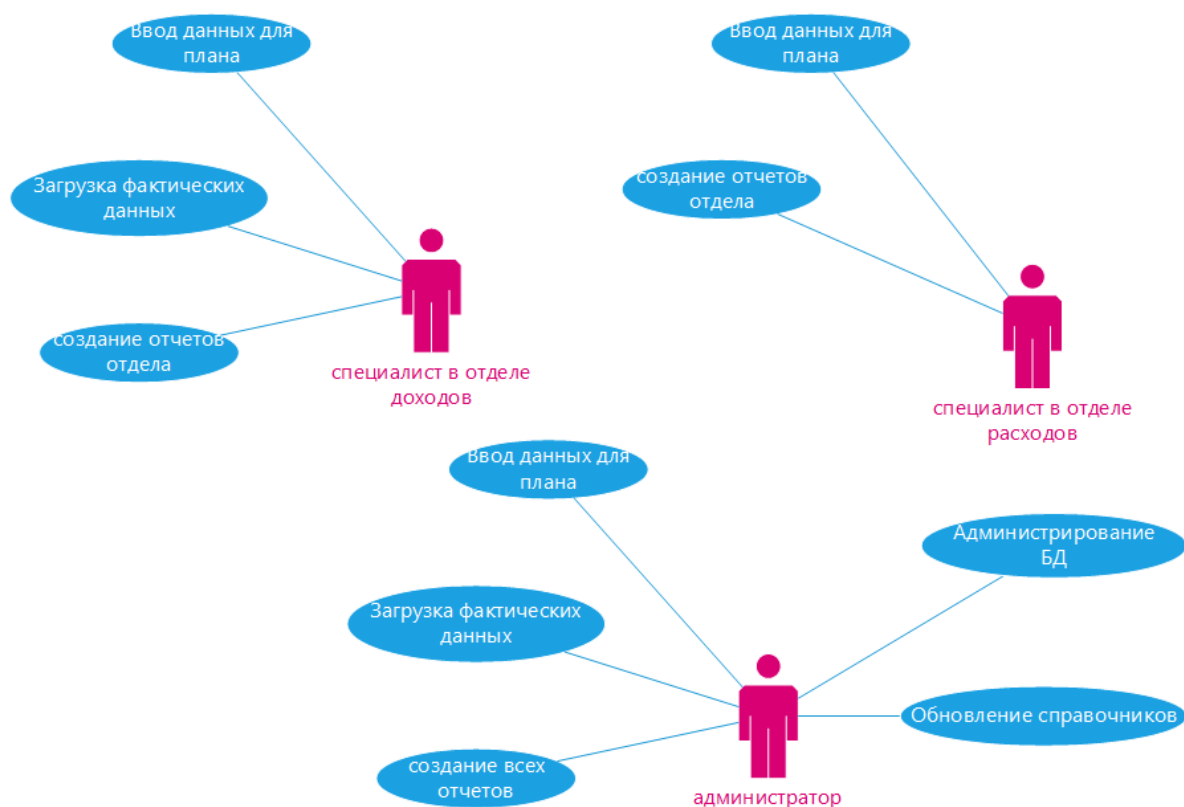


Рисунок 1.4 – Диаграмма прецедентов

На рисунке 1.4 показано какие привилегии есть у подгрупп пользователей: специалист в отделе расходов, специалист в отделе доходов, администратор(таблица 1.1).

Таблица 1.1 – Подгруппа пользователей

№	Описание требований	Субъект	Прецедент
1	Специалист по доходам должен вводить уточнения и бюджет следующего года по своему отделу	Специалист по доходам	Ввод данных для плана
2	Специалист по доходам должен загружать ежедневные поступления с Казначейства	Специалист по доходам	Загрузка фактических данных
3	Специалист по доходам должен создавать отчеты	Специалист по доходам	Создание отчетов отдела
4	Специалист по расходам должен вводить уточнения и бюджет следующего года по своему отделу	Специалист по расходам	Ввод данных для плана
5	Специалист по расходам должен создавать отчеты	Специалист по расходам	Создание отчетов отдела
6	Администратор должен создавать резервную копию базы данных	Администратор	Администрирование БД

7	Администратор должен обновлять справочники до актуальных значений	Администратор	Обновление справочников
8	Администратор может помочь в загрузке данных	Администратор	Загрузка фактических данных
9	Администратор может помочь в вводе утверждений в любом отделе	Администратор	Ввод данных плана

Диаграмма развертывания

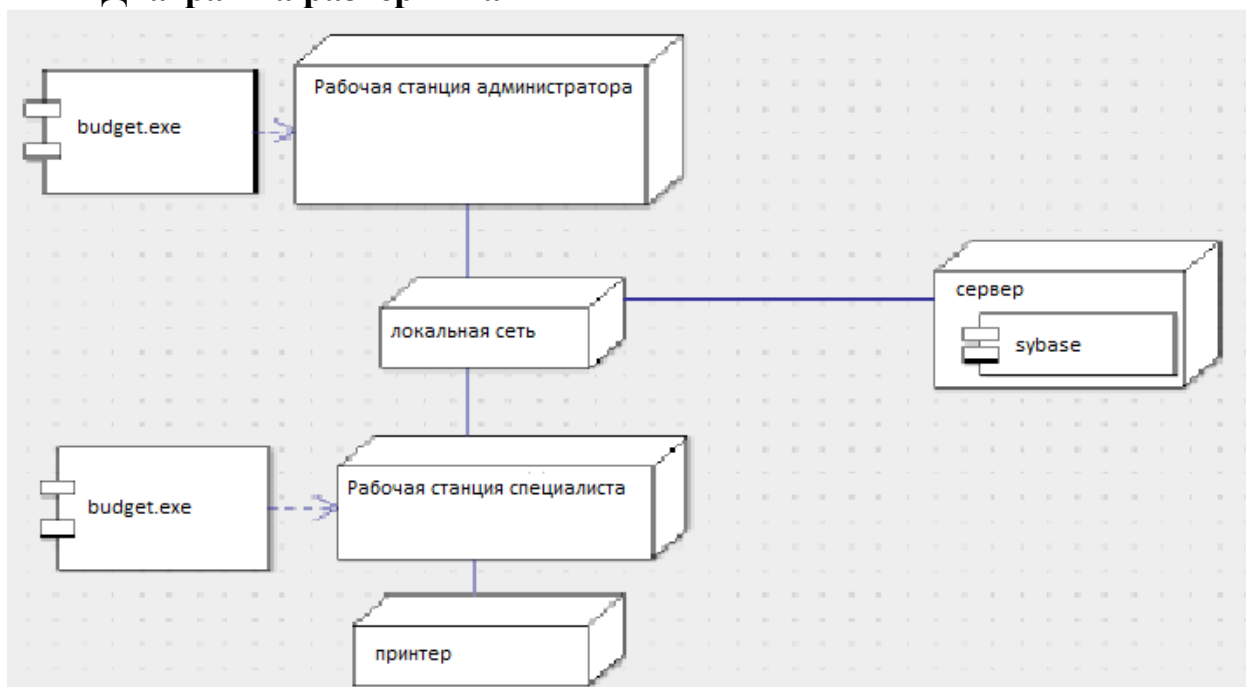


Рисунок 1.5 – Диаграмма развертывания

Узлы показывают, как прямоугольные с артефактами, расположенными в их, изображенными в стиле прямоугольников. Узлы имеют все шансы обладать под узлами, которые представляется как вложенные прямоугольники.

Узел диаграммы развертывания имеет возможность концептуально изображать множество физических узлов, таких как сервера БД.

Существуют две типы узлов:

- узел устройств;
- узел среды исполнения.

Узлы устройств считаются физически вычислительными ресурсами со своей собственной памятью и сервисами для исполнения программного обеспечения, такие как обыденные ПК, мобильные телефонные аппараты. Узел среды выполнения – это программный вычислительный ресурс, который действует внутри внешнего узла и который предоставляет собой сервис, выполняющий другие исполняемые программные элементы.

Диаграмма развертывания нашей системы очень проста, так как не требует специального оборудования и может устанавливаться на любой ПК.

1.3 Проектирование базы данных

1.3.1 Определение атрибутов всех сущностей

В таблице 1.2 представлены атрибуты каждой из сущностей созданной базы данных. Предоставленная таблица нужна нам для того, чтоб продолжить дальнейшее проектирование базы данных. В ней определяются все атрибуты, которые будут в таблицах, а также приводится их краткое описание.

Таблица 1.2 – Таблицы БД

action_eduCultur мероприятия		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
id_action		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
Actual		varchar(1)
develop		varchar(1)
action_GosApparat мероприятия		
RecordID		integer(16)
fgroup		varchar(2)

fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
id_action		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)
action_RealSector мероприятия		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
id_action		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)

Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)
action_Social мероприятия		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
id_action		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)

year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)
b_org организации		
Code		varchar(10)
OrgName		varchar(150)
Short_OrgName		varchar(100)
BankCode		varchar(20)
BankName		varchar(50)
BankLocation		varchar(20)
Bill		varchar(20)
RNN		varchar(20)
RecordID		integer(16)
Banks банки		
RecordID		integer(4)
Code		varchar(10)
BankName		varchar(100)
Corchet		varchar(20)
City		varchar(30)
Corr_date		datetime(8)
Notice		varchar(400)
budget_incomes доходы		
Category		varchar(1)
Class		varchar(2)
Subclass		varchar(1)
Specifique		varchar(2)
EndDate		datetime(8)
RussianName		varchar(400)
KazakhName		varchar(400)
Plan_Total		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Difference		Money(8)
addition		Money(8)
Revision		Money(8)
corrected_budget		Money(8)
year_		smallint(2)
versionNumber		smallint(2)
departmentCode		varchar(100)

corr_date		datetime(8)
actual		varchar(1)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
RecordID		integer(16)
Revision_date		datetime(8)
budget_outcomes		
расходы		
fgroup		varchar(10)
fsubgroup		varchar(10)
abp		varchar(10)
program		varchar(10)
subprogram		varchar(50)
specifique		varchar(10)
endDate		datetime(8)
RussianName		varchar(400)
KazakhName		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Difference		Money(8)
Addition		Money(8)
Revision		Money(8)
Corrected_budget		Money(8)
year_		smallint(2)
versionNumber		smallint(2)
departmentCode		varchar(100)
RecordID		integer(16)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
corr_date		datetime(8)
Revision_date		datetime(8)
actual		varchar(1)
develop		varchar(1)

daily_income		
платежки		
Category		varchar(1)
Class		varchar(2)
Specifique		varchar(2)
RussianName		varchar(400)
KazakhName		varchar(400)
NPP		varchar(100)
Payer		varchar(100)
Payer_RNN		varchar(20)
PaySum		Money(8)
Receiver		varchar(100)
R_RNN		varchar(20)
pay_date		datetime(8)
Subclass		varchar(1)
RecordID		integer(16)
daily_income_sum		
платежки агрегир		
Category		varchar(4)
Class		varchar(4)
Subclass		varchar(4)
Specifique		varchar(4)
summ		Money(8)
pay_date		datetime(8)
RecordID		integer(16)
dep_fin_incomes		
депФин ДОХОДЫ		
RecordID		integer(16)
kod		varchar(4)
Category		varchar(3)
Class		varchar(3)
Subclass		varchar(3)
Specifique		varchar(3)
naim		varchar(400)
plan_		Money(8)
Utoch		Money(8)
Corrected		Money(8)
by_payment		Money(8)
by_obligation		Money(8)
sum_permit		Money(8)
cash_exec		Money(8)
date_		datetime(8)

dep_fin_outcomes		
депФин расходы		
kod		varchar(4)
fgroup		varchar(3)
subgroup		varchar(3)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
naim		varchar(400)
plan_		Money(8)
Utoch		Money(8)
Corrected		Money(8)
by_payment		Money(8)
by_obligation		Money(8)
sum_permit		Money(8)
cash_exec		Money(8)
date_		datetime(8)
expenses_eduCultur		
рабочая		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
endDate		datetime(8)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)

Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
departmentCode		varchar(10)
develop		varchar(1)
actual		varchar(1)
expenses_GosApparat		
рабочая		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
endDate		datetime(8)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
departmentCode		varchar(10)
develop		varchar(1)
actual		varchar(1)
expenses_RealSector		
рабочая		

RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
endDate		datetime(8)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
departmentCode		varchar(10)
develop		varchar(1)
actual		varchar(1)
expenses_Social		
рабочая		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
endDate		datetime(8)
naimRus		varchar(400)
naimKaz		varchar(400)

Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
departmentCode		varchar(10)
develop		varchar(1)
actual		varchar(1)
GatheringIncomes		
Собираемость доходов		
RecordID		integer(4)
Month_		varchar(2)
year_		smallint(2)
Plan_		Money(8)
Fact_		Money(8)
corr_date		datetime(8)
version		smallint(2)
actual		varchar(1)
LoansMain		
Займы - проект		
idLoan		integer(4)
naimRus		varchar(250)
naimKaz		varchar(250)
credRus		varchar(250)
credKaz		varchar(250)
currency		varchar(50)
osn		Money(8)
yearFrom		smallint(2)
yearTo		smallint(2)
actual		varchar(1)

vozm		Money(8)
vozn		Money(8)
rate		varchar(10)
loansTransh займы – каждый год проекта		
idLoan		integer(4)
year_		smallint(2)
numTransh		smallint(2)
osn_t		Money(8)
vozm_t		Money(8)
vozn_t		Money(8)
penya_t		Money(8)
osn_d		Money(8)
vozm_d		Money(8)
vozn_d		Money(8)
penya_d		Money(8)
date_		datetime(8)
pogasheno		varchar(1)
actual		varchar(1)
loansYear займы каждый транш		
idLoan		integer(4)
year_		smallint(2)
penya		integer(16)
osn_t		Money(8)
vozm_t		Money(8)
vozn_t		Money(8)
penya_t		Money(8)
osn_d		Money(8)
vozm_d		Money(8)
vozn_d		Money(8)
penya_d		Money(8)
u_osn_t		Money(8)
u_vozm_t		Money(8)
u_vozn_t		Money(8)
u_penya_t		Money(8)
u_osn_d		Money(8)
u_vozm_d		Money(8)
u_vozn_d		Money(8)
u_penya_d		Money(8)
actual		varchar(1)
sp_departments		
dep_code		varchar(10)

RussianName		varchar(150)
KazakhName		varchar(150)
RusName		varchar(150)
KazName		varchar(150)
RecordID		integer(16)
sp_incomes_category		
Category		varchar(1)
RussianName		varchar(255)
KazakhName		varchar(255)
RecordID		integer(16)
date_		datetime(8)
actual		varchar(1)
year_		smallint(2)
sp_incomes_class		
Category		varchar(1)
Class		varchar(2)
RussianName		varchar(255)
KazakhName		varchar(255)
RecordID		integer(16)
date_		datetime(8)
actual		varchar(1)
year_		smallint(2)
sp_incomes_specifique		
Имя		Тип(Размер)
Category		varchar(1)
Class		varchar(2)
Subclass		varchar(1)
Specifique		varchar(2)
RussianName		varchar(255)
KazakhName		varchar(255)
RecordID		integer(16)
date_		datetime(8)
actual		varchar(1)
year_		smallint(2)
sp_incomes_subclass		
Category		varchar(1)
Class		varchar(2)
Subclass		varchar(1)
RussianName		varchar(255)
KazakhName		varchar(255)
RecordID		integer(16)
date_		datetime(8)
actual		varchar(1)

year_		smallint(2)
sp_outcomes_abp		
Code		varchar(3)
DistrictCode		varchar(10)
RussianName		varchar(250)
KazakhName		varchar(250)
Corr_date		datetime(8)
Notice		varchar(400)
RecordID		integer(16)
date_		datetime(8)
actual		varchar(1)
year_		smallint(2)
sp_outcomes_group		
fgroup		varchar(2)
RussianName		varchar(255)
KazakhName		varchar(255)
depCode		varchar(10)
RecordID		integer(16)
date_		datetime(8)
actual		varchar(1)
year_		smallint(2)
sp_outcomes_program		
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
RussianName		varchar(255)
KazakhName		varchar(255)
actual		varchar(1)
date_		datetime(8)
depCode		varchar(10)
sp_outcomes_specif		
Category		varchar(1)
Class		varchar(1)
Specifique		varchar(3)
RussianName		varchar(255)
KazakhName		varchar(255)
RecordID		integer(16)
date_		datetime(8)
actual		varchar(1)
year_		smallint(2)
sp_outcomes_subgroup		
fgroup		varchar(2)

fsubgroup		vchar(1)
RussianName		vchar(255)
KazakhName		vchar(255)
depCode		vchar(10)
RecordID		integer(16)
date_		datetime(8)
actual		vchar(1)
year_		smallint(2)
sp_outcomes_subprogram		
fgroup		vchar(2)
fsubgroup		vchar(1)
abp		vchar(3)
program		vchar(3)
subprogram		vchar(3)
RussianName		vchar(255)
KazakhName		vchar(255)
actual		vchar(1)
date_		datetime(8)
depCode		vchar(10)
specif1_eduCultur		
RecordID		integer(16)
fgroup		vchar(2)
fsubgroup		vchar(1)
abp		vchar(3)
program		vchar(3)
subprogram		vchar(3)
specifique		vchar(3)
id_specif1		integer(4)
naimRus		vchar(400)
naimKaz		vchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)

Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
Actual		varchar(1)
Develop		varchar(1)
specif1_GosApparat		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
id_specif1		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)
specif1_RealSector		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)

program		vchar(3)
subprogram		vchar(3)
specifique		vchar(3)
id_specif1		integer(4)
naimRus		vchar(400)
naimKaz		vchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
actual		vchar(1)
develop		vchar(1)
specif1_Social		
RecordID		integer(16)
fgroup		vchar(2)
fsubgroup		vchar(1)
abp		vchar(3)
program		vchar(3)
subprogram		vchar(3)
specifique		vchar(3)
id_specif1		integer(4)
naimRus		vchar(400)
naimKaz		vchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)

Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)
specif2_eduCultur		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
action		integer(4)
specif1		integer(4)
id_specif2		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)

year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)
specif2_GosApparat		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
action		integer(4)
specif1		integer(4)
id_specif2		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)
specif2_RealSector		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)

subprogram		varchar(3)
specifique		varchar(3)
action		integer(4)
specif1		integer(4)
id_specif2		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)
specif2_Social		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
action		integer(4)
specif1		integer(4)
id_specif2		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)

Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)
Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)
typeOfWork_RealSector		
RecordID		integer(16)
fgroup		varchar(2)
fsubgroup		varchar(1)
abp		varchar(3)
program		varchar(3)
subprogram		varchar(3)
specifique		varchar(3)
action		integer(4)
id_typeOfWork		integer(4)
naimRus		varchar(400)
naimKaz		varchar(400)
volumeOfWork		varchar(255)
Plan_		Money(8)
Plan_MB		Money(8)
Plan_RB		Money(8)
Transfert		Money(8)
Transfert_MB		Money(8)
Transfert_RB		Money(8)
Difference		Money(8)
Difference_MB		Money(8)
Difference_RB		Money(8)
Addition		Money(8)
Revision		Money(8)
Revision_MB		Money(8)
Revision_RB		Money(8)

Revision_date		datetime(8)
Corrected_budget		Money(8)
corr_date		datetime(8)
year_		smallint(2)
versionNumber		smallint(2)
actual		varchar(1)
develop		varchar(1)

1.4 Используемые технологии программирования

Для основной цели нашего проекта можно использовать разные виды программирования. Такие как языки для прикладной программы (Delphi, C#, C++, Java и т.п.) или же языки для Web-систем (PHP, Java, .Net, Ruby и т.п.) и так же разные виды СУБД (Sybase, MySQL, MSSQL, Oracle и т.п.). Далее я объясню почему я использовал именно основную связку Sybase+Delphi.

1.4.1 СУБД Sybase (ASE)

Adaptive Server Enterprise (ASE) — высокопроизводительная система управления реляционными базами данных для критически важных сред интенсивной обработки данных. Система обеспечивает высочайшие производственную эффективность и пропускную способность, поддерживая широкий спектр платформ.

Sybase ASE является не такая распространённая как другие СУБД (такие как Oracle, MySQL, MSSQL), но является одной из давних СУБД и на ее основе сделали MSSQL. В следствии того что Sybase не так распространён его реже взламывают по SAP ASE Total Cost of Ownership Sybase обеспечивает бескомпромиссную надежность, превосходную производительность, при этом Sybase дешевле чем Oracle.

Так же являясь программным продуктом много миллиардной немецкой компании SAP Sybase получает крупную поддержку и совершенствуется не прерывно.

1.4.2 Delphi 7, Perl

Причина почему был использован прикладной язык программирования, а не набирающий популярность Web-программирование, это закрытость системы от входа. Так даже если у вас есть сеть к серверу вы не сможете зайти туда без прикладной программы. Delphi был выбран по хорошей совместимости с СУБД Sybase.

Perl был использован для интеграции с другими проектами. Perl является универсальным языком, а так же Perl входит во многие Linux серверы как стандартный пакет.

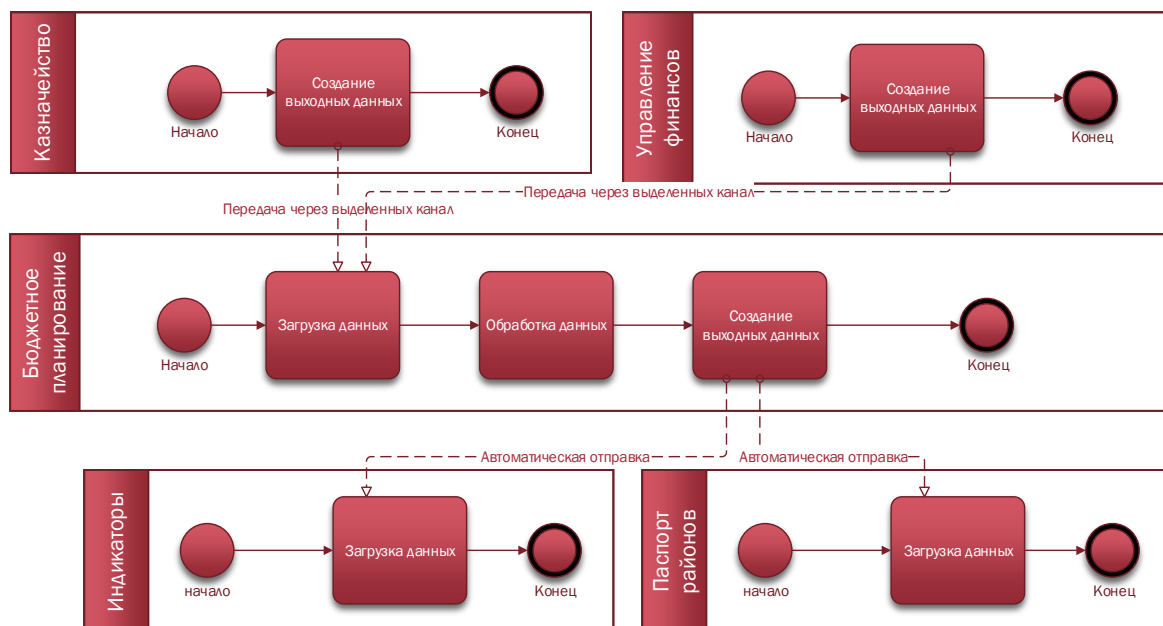


Рисунок 1.6 Диаграмма интеграции с другими проектами

2 Расчетная часть

2.1 Расчет места для хранения базы данных

В данном шаге, нужно знать какой объем памяти станет занимать создаваемая БД. Объём наружной памяти, нужный для функционирования системы, формируется из двух составляющих: память, занимаемая модулями СУБД и память, отводимая под данные (M_d). Более существенным обычно является M_d . Размер памяти, занимаемый программными модулями юзера, традиционно слишком невелик по сравнению с объёмом самих данных, поэтому имеет возможность не учитываться. Так как в БД есть похожие таблицы посчитаем усреднено их значение. В работе рассчитывается ориентировочный наибольший объем памяти занимаемой Базы Данных. Расчет физической памяти таблиц указана в таблицах 2.1-2.20

Таблица 2.1- Подсчет физической памяти для action_eduCultur

action_eduCultur мероприятия		
Имя поля	Длина	Тип данных
RecordID	16	integer(16)
fgroup	2	varchar(2)
fsubgroup	1	varchar(1)
abp	3	varchar(3)
program	3	varchar(3)
subprogram	3	varchar(3)
specifique	3	varchar(3)
id_action	4	integer(4)
naimRus	400	varchar(400)
naimKaz	400	varchar(400)
Plan_	8	Money(8)
Plan_MB	8	Money(8)

Plan_RB	8	Money(8)
Transfert	8	Money(8)
Transfert_MB	8	Money(8)
Transfert_RB	8	Money(8)
Difference	8	Money(8)
Difference_MB	8	Money(8)
Difference_RB	8	Money(8)
Addition	8	Money(8)
Revision	8	Money(8)
Revision_MB	8	Money(8)
Revision_RB	8	Money(8)
Revision_date	8	datetime(8)
Corrected_budget	8	Money(8)
corr_date	8	datetime(8)
year_	2	smallint(2)
versionNumber	2	smallint(2)
Actual	1	varchar(1)
develop	1	varchar(1)

Общий объем поля: 969 байт

Количество: ~ 15 000

Требуемая память: 14 535 000 байт

Таблица 2.2 - Подсчет физической памяти для b_org

b_org организации		
Имя поля	Длина	Тип данных
Code	10	varchar(10)
OrgName	150	varchar(150)
Short_OrgName	100	varchar(100)
BankCode	20	varchar(20)
BankName	50	varchar(50)
BankLocation	20	varchar(20)
Bill	20	varchar(20)
RNN	20	varchar(20)
RecordID	16	integer(16)

Общий объем поля: 406 байт

Количество: ~ 50

Требуемая память: 20 300 байт

Таблица 2.3 - Подсчет физической памяти для Banks

Banks банки		
Имя поля	Длина	Тип данных

RecordID	4	integer(4)
Code	10	varchar(10)
BankName	100	varchar(100)
Corchet	20	varchar(20)
City	30	varchar(30)
Corr_date	8	datetime(8)
Notice	400	varchar(400)

Общий объем поля: 572 байт

Количество: ~ 50

Требуемая память: 28 600 байт

Таблица 2.4 - Подсчет физической памяти для budget_incomes

budget_incomes		
доходы		
Имя поля	Длина	Тип данных
Category	1	varchar(1)
Class	2	varchar(2)
Subclass	1	varchar(1)
Specifique	2	varchar(2)
EndDate	8	datetime(8)
RussianName	400	varchar(400)
KazakhName	400	varchar(400)
Plan_Total	8	Money(8)
Plan_MB	8	Money(8)
Plan_RB	8	Money(8)
Transfert	8	Money(8)
Difference	8	Money(8)
addition	8	Money(8)
Revision	8	Money(8)
corrected_budget	8	Money(8)
year_	2	smallint(2)
versionNumber	2	smallint(2)
departmentCode	100	varchar(100)
corr_date	8	datetime(8)
actual	1	varchar(1)
Transfert_MB	8	Money(8)
Transfert_RB	8	Money(8)
Difference_MB	8	Money(8)
Difference_RB	8	Money(8)
Revision_MB	8	Money(8)
Revision_RB	8	Money(8)
RecordID	16	integer(16)

Revision_date	8	datetime(8)
---------------	---	-------------

Общий объем поля: 1063 байт

Количество: ~ 25 000

Требуемая память: 26 575 000 байт

Таблица 2.5 - Подсчет физической памяти для budget_outcomes

budget_outcomes		
расходы		
Имя поля	Длина	Тип данных
fgroup	10	varchar(10)
fsubgroup	10	varchar(10)
abp	10	varchar(10)
program	10	varchar(10)
subprogram	50	varchar(50)
specifique	10	varchar(10)
endDate	8	datetime(8)
RussianName	400	varchar(400)
KazakhName	400	varchar(400)
Plan_	8	Money(8)
Plan_MB	8	Money(8)
Plan_RB	8	Money(8)
Transfert	8	Money(8)
Difference	8	Money(8)
Addition	8	Money(8)
Revision	8	Money(8)
Corrected_budget	8	Money(8)
year_	2	smallint(2)
versionNumber	2	smallint(2)
departmentCode	100	varchar(100)
RecordID	16	integer(16)
Transfert_MB	8	Money(8)
Transfert_RB	8	Money(8)
Difference_MB	8	Money(8)
Difference_RB	8	Money(8)
Revision_MB	8	Money(8)
Revision_RB	8	Money(8)
corr_date	8	datetime(8)
Revision_date	8	datetime(8)
actual	1	varchar(1)
develop	1	varchar(1)

Общий объем поля: 1153 байт

Количество: ~ 50 000

Требуемая память: 57 900 000байт

Таблица 2.6 - Подсчет физической памяти для daily_income

daily_income платежки		
Имя поля	Длина	Тип данных
Category	1	varchar(1)
Class	2	varchar(2)
Subclass	4	varchar(4)
Specifique	2	varchar(2)
RussianName	400	varchar(400)
KazakhName	400	varchar(400)
NPP	100	varchar(100)
Payer	100	varchar(100)
Payer_RNN	20	varchar(20)
PaySum	8	Money(8)
Receiver	100	varchar(100)
R_RNN	20	varchar(20)
pay_date	8	datetime(8)
RecordID	16	integer(16)

Общий объем поля: 1181 байт

Количество: ~ 50 000

Требуемая память: 59 050 000байт

Таблица 2.7 - Подсчет физической памяти для daily_income_sum

daily_income_sum платежки агрегир		
Имя поля	Длина	Тип данных
Category	4	varchar(4)
Class	4	varchar(4)
Subclass	4	varchar(4)
Specifique	4	varchar(4)
summ	8	Money(8)
pay_date	8	datetime(8)
RecordID	16	integer(16)

Общий объем поля: 48 байт

Количество: ~ 50 000

Требуемая память: 2 400 000байт

Таблица 2.8 - Подсчет физической памяти для dep_fin_incomes

dep_fin_incomes		
депФин доходы		
Имя поля	Длина	Тип данных
RecordID	16	integer(16)
kod	4	varchar(4)
Category	3	varchar(3)
Class	3	varchar(3)
Subclass	3	varchar(3)
Specifique	3	varchar(3)
naim	400	varchar(400)
plan_	8	Money(8)
Utoch	8	Money(8)
Corrected	8	Money(8)
by_payment	8	Money(8)
by_obligation	8	Money(8)
sum_permit	8	Money(8)
cash_exec	8	Money(8)
date_	8	datetime(8)

Общий объем поля: 496 байт

Количество: ~ 15 000

Требуемая память: 7 440 000байт

Таблица 2.9 - Подсчет физической памяти для expenses_educultur

expenses_educultur		
рабочая		
Имя поля	Длина	Тип данных
RecordID	16	integer(16)
fgroup	2	varchar(2)
fsubgroup	1	varchar(1)
abp	3	varchar(3)
program	3	varchar(3)
subprogram	3	varchar(3)
specifique	3	varchar(3)
endDate	8	datetime(8)
naimRus	400	varchar(400)
naimKaz	400	varchar(400)
Plan_	8	Money(8)
Plan_MB	8	Money(8)
Plan_RB	8	Money(8)
Transfert	8	Money(8)
Transfert_MB	8	Money(8)
Transfert_RB	8	Money(8)

Difference	8	Money(8)
Difference_MB	8	Money(8)
Difference_RB	8	Money(8)
Addition	8	Money(8)
Revision	8	Money(8)
Revision_MB	8	Money(8)
Revision_RB	8	Money(8)
Revision_date	8	datetime(8)
Corrected_budget	8	Money(8)
corr_date	8	datetime(8)
year_	2	smallint(2)
versionNumber	2	smallint(2)
departmentCode	10	varchar(10)
develop	1	varchar(1)
actual	1	varchar(1)

Общий объем поля: 983 байт

Количество: ~ 15 000

Требуемая память: 14 745 000байт

Таблица 2.10 - Подсчет физической памяти для sp_departments

sp_departments		
Имя поля	Длина	Тип данных
dep_code	10	varchar(10)
RussianName	150	varchar(150)
KazakhName	150	varchar(150)
RusName	150	varchar(150)
KazName	150	varchar(150)
RecordID	16	integer(16)

Общий объем поля: 626 байт

Количество: ~ 8

Требуемая память: 5 008байт

Таблица 2.12 - Подсчет физической памяти для sp_incomes_category

sp_incomes_category		
Имя поля	Длина	Тип данных
Category	1	varchar(1)
RussianName	255	varchar(255)
KazakhName	255	varchar(255)
RecordID	16	integer(16)
date_	8	datetime(8)
Actual	1	varchar(1)

year_	2	smallint(2)
-------	---	-------------

Общий объем поля: 538 байт
Количество: ~ 8
Требуемая память: 4 304 байт

Таблица 2.13 - Подсчет физической памяти для sp_incomes_class

sp_incomes_class		
Имя поля	Длина	Тип данных
Category	1	varchar(1)
Class	2	varchar(2)
RussianName	255	varchar(255)
KazakhName	255	varchar(255)
RecordID	16	integer(16)
date_	8	datetime(8)
Actual	1	varchar(1)
year_	2	smallint(2)

Общий объем поля: 540 байт
Количество: ~ 64
Требуемая память: 34 560 байт

Таблица 2.14 - Подсчет физической памяти для sp_incomes_specifique

sp_incomes_specifique		
Имя поля	Длина	Тип данных
Category	1	varchar(1)
Class	2	varchar(2)
Subclass	1	varchar(1)
Specifique	2	varchar(2)
RussianName	255	varchar(255)
KazakhName	255	varchar(255)
RecordID	16	integer(16)
date_	8	datetime(8)
Actual	1	varchar(1)
year_	2	smallint(2)

Общий объем поля: 543 байт
Количество: ~ 300
Требуемая память: 162 900 байт

Таблица 2.15 - Подсчет физической памяти для sp_incomes_subclass

sp_incomes_subclass		
Имя поля	Длина	Тип данных

Category	1	varchar(1)
Class	2	varchar(2)
Subclass	1	varchar(1)
RussianName	255	varchar(255)
KazakhName	255	varchar(255)
RecordID	16	integer(16)
date_	8	datetime(8)
Actual	1	varchar(1)
year_	2	smallint(2)

Общий объем поля: 541 байт

Количество: ~ 180

Требуемая память: 97 380 байт

Таблица 2.16 - Подсчет физической памяти для sp_outcomes_abp

sp_outcomes_abp		
Имя поля	Длина	Тип данных
Code	3	varchar(3)
DistrictCode	10	varchar(10)
RussianName	250	varchar(250)
KazakhName	250	varchar(250)
Corr_date	8	datetime(8)
Notice	400	varchar(400)
RecordID	16	integer(16)
date_	8	datetime(8)
Actual	1	varchar(1)
year_	2	smallint(2)

Общий объем поля: 948 байт

Количество: ~ 50

Требуемая память: 47 400 байт

Таблица 2.17 - Подсчет физической памяти для sp_outcomes_group

sp_outcomes_group		
Имя поля	Длина	Тип данных
Fgroup	2	varchar(2)
RussianName	255	varchar(255)
KazakhName	255	varchar(255)
depCode	10	varchar(10)
RecordID	16	integer(16)
date_	8	datetime(8)
actual	1	varchar(1)
year_	2	smallint(2)

Общий объем поля: 549 байт
 Количество: ~ 13
 Требуемая память: 7 137 байт

Таблица 2.18 - Подсчет физической памяти для sp_outcomes_program

sp_outcomes_program		
Имя поля	Длина	Тип данных
fgroup	2	varchar(2)
fsubgroup	1	varchar(1)
abp	3	varchar(3)
program	3	varchar(3)
RussianName	255	varchar(255)
KazakhName	255	varchar(255)
actual	1	varchar(1)
date_	8	datetime(8)
depCode	10	varchar(10)

Общий объем поля: 538 байт
 Количество: ~ 30
 Требуемая память: 16 140 байт

Таблица 2.19 - Подсчет физической памяти для sp_outcomes_specif

sp_outcomes_specif		
Имя поля	Длина	Тип данных
Category	1	varchar(1)
Class	1	varchar(1)
Specifique	3	varchar(3)
RussianName	255	varchar(255)
KazakhName	255	varchar(255)
RecordID	16	integer(16)
date_	8	datetime(8)
actual	1	varchar(1)
year_	2	smallint(2)

Общий объем поля: 542 байт
 Количество: ~ 300
 Требуемая память: 162 600 байт

Таблица 2.19 - Подсчет физической памяти для sp_outcomes_subgroup

sp_outcomes_subgroup		
Имя поля	Длина	Тип данных
fgroup	2	varchar(2)

fsubgroup	1	varchar(1)
RussianName	255	varchar(255)
KazakhName	255	varchar(255)
depCode	10	varchar(10)
RecordID	16	integer(16)
date_	8	datetime(8)
actual	1	varchar(1)
year_	2	smallint(2)

Общий объем поля: 550 байт
Количество: ~ 150
Требуемая память: 82 500 байт

Таблица 2.20 - Подсчет физической памяти для sp_outcomes_subprogram

sp_outcomes_subprogram		
Имя поля	Длина	Тип данных
fgroup	2	varchar(2)
fsubgroup	1	varchar(1)
abp	3	varchar(3)
program	3	varchar(3)
subprogram	3	varchar(3)
RussianName	255	varchar(255)
KazakhName	255	varchar(255)
actual	1	varchar(1)
date_	8	datetime(8)
depCode	10	varchar(10)

Общий объем поля: 541 байт
Количество: ~ 64
Требуемая память: 34 624 байт

Итого: $M_c = 34624 + 82500 + 162600 + 16140 + 7137 + 47400 + 97380 + 162900 + 34560 + 4304 + 5008 + 4*(1474500) + 7440000 + 2400000 + 59050000 + 57900000 + 26575000 + 28600 + 20300 + 1453500*3 = 256653453$ байт = 256Мб.

Объем таблиц колоссальный это видно по расчету на 1 год. Если рассчитать на 5 или более лет то мы получим примерно 1.2 гигабайта данных. Для дальнейшей реализации создана архивация данных в прикладной программе.

2.2 Меры по обеспечению безопасности

Есть много разных обеспечений безопасности. В моем случае было сделан акцент на изолированный сервер в акимате без доступа к интернету,

так же были прописаны IP шлюзы которые могут заходить в данный сервер для ограничения в не санкционированном входе из вне Управления экономики и бюджетного планирования.

3 Разработка ПО

3.1 Разработка интерфейса ИС «Бюджетное планирование»

Данное приложение разработана с учетом интуитивно-понятного интерфейса.

С помощью моего приложения менеджерам будет проще вводить информацию, а клиенты проще взаимодействовать с менеджерами. На рисунке 3.1 форма авторизации пользователя.

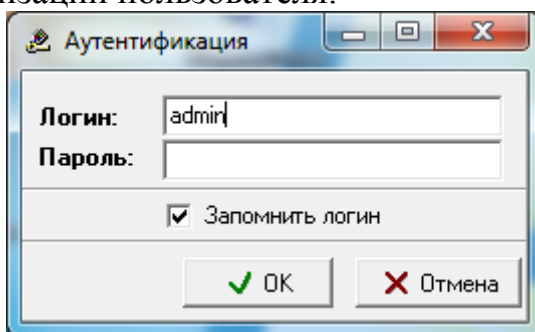


Рисунок 3.1 – Форма авторизации

Интуитивно понятный интерфейс – это интерфейс пользователя, который должен комфортно получать информацию, он не должен тратить свое лишнее время при поиске. В интерфейсе должно использоваться исключительно простые формы и мягкие расцветки. Это позволяет понизить нагрузки на глаза юзера и концентрировать внимание того, кто пользуется системой.

Для всех категорий пользователей, работа в программе начинается с авторизации. В программе предвидено разделение на три группы пользователей по уровню доступа: специалист по доходам, администратор, специалист по расходам.

Далее пользователя ждет главная форма. Смотря под кем пользователь зашел его ждет разные кнопки. На пример у администратора видны все отделы а так же кнопка «Администрирования».



Рисунок 3.2 – главная форма

Форма ввода данных для отделов расхода представляются идентичный вид. Меняется подключение к разным таблицам. В данной форме есть две основные рабочие области.

1. Слева данные сортированные по ЕБК так же они могут свернуты для удобства. В данной области находится данные бюджета заданного бюджетного года. Так же Уточнения на последний квартал.

Данные				Настройка	Запись	Взаимодействие с Excel													
ОГ	ОГ	АП	П	ПП	С	Ра	Наименование	Утвержденный бюджет	РБ	МБ	Трансферты РБ	Дополнение	Уточненный	Уточненный РБ	Уточненный МБ	Скорректирован	Скорр.РБ	Скорр.МБ	
							Текущие												
							Развития												
01							Государственные услуги общего характера	1 000 000,00	80 000,00	1 000 000,00	0,00	0,00	0 000 000,00	0 000 000,00	1 000 000,00	0,00	0,00	0,00	
1							Представительные, исполнительные и другие органы, выполняющие общие функции государственного управления	1 000 000,00	80 000,00	1 000 000,00	0,00	0,00	0 000 000,00	0 000 000,00	1 000 000,00	0,00	0,00	0,00	
	111						Аппарат наместника города республиканского значения, столицы	10 000,00	1 700,00	10 000,00	0,00	0,00	10 000,00	10 000,00	10 000,00	0,00	0,00	0,00	
		001					Услуги по обеспечению деятельности наместника города республиканского значения, столицы	10 000,00	1 700,00	10 000,00	0,00	0,00	10 000,00	10 000,00	10 000,00	0,00	0,00	0,00	
			011				За счет трансфертов из республиканского бюджета	1 700,00	1 700,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
				111			Оплата труда	200,00	200,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
				112			Компенсационные выплаты	200,00	200,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
				121			Социальный налог	10,00	10,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
				122			Социальные отчисления в Государственный фонд социального страхования	10,00	10,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
				131			Оплата труда технического персонала	80,00	80,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
				135			Взносы работодателей по техническому персоналу	20,00	20,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
				159			Оплата прочих услуг и работ	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	0,00	
			015				За счет средств местного бюджета	10 000,00	0,00	10 000,00	0,00	0,00	10 000,00	10 000,00	10 000,00	0,00	0,00	0,00	
		003					Капитальные расходы государственного органа	800,00	0,00	800,00	0,00	0,00	800,00	0,00	800,00	0,00	0,00	0,00	
	121						Аппарат акция города республиканского значения, столицы	1 000 000,00	0 000 000,00	1 000 000,00	0,00	0 000 000,00	1 000 000,00	1 000 000,00	0 000 000,00	0,00	0,00	0,00	
	123						Аппарат акция района в городе, города районного значения, поселка, аула (села), аульского (сельского) округа	200 000,00	0 000 000,00	200 000,00	0,00	0 000 000,00	200 000,00	0 000 000,00	0 000 000,00	0,00	0,00	0,00	
	379						Ревизионная комиссия города республиканского значения, столицы	100 000,00	0 000 000,00	100 000,00	0,00	0 000 000,00	100 000,00	0 000 000,00	0 000 000,00	0,00	0,00	0,00	
2							Финансовая деятельность	100 000,00	0 000 000,00	100 000,00	0,00	0 000 000,00	100 000,00	0 000 000,00	0 000 000,00	0,00	0,00	0,00	
5							Планирование и статистическая деятельность	100 000,00	0 000 000,00	100 000,00	0,00	0 000 000,00	100 000,00	0 000 000,00	0 000 000,00	0,00	0,00	0,00	

Рисунок 3.3 – Форма расходов

2. Справа панель настроек в ней находится основной функционал формы. В данной панели размещены условия показа данных далее дата за какую дату брать уточнение, галочки для разворачивания полной таблицы, кнопка «Показать» показывает таблицу на заданную дату, кнопка «Сохранить» сохраняет все изменения в таблице так же пересчитывает данные, кнопки +С и –ПП являются динамическими кнопками и меняются, смотря где находится курсор и нужны для создания подразделов или удаления выше стоящих разделов, кнопка «Восстановить» возвращает удаленные разделы. Кнопки «Проект бюджета» и «Уточнение» создают проект бюджета на заданный год или уточнения на заданную дату, эти кнопки инактивные у всех кроме сводного отдела и администратора.

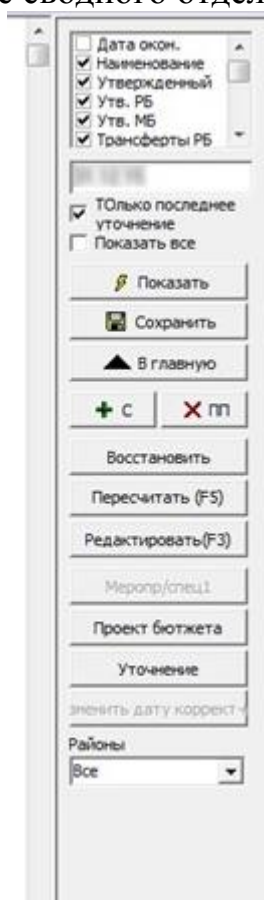


Рисунок 3.4 –форма расходов

Форма для отдела доходов не много отличается от расходов. Левая сторона отличается меньшей разветвленностью так как ЕБК у доходов состоит из 4 разделов (категория, класс, субкласс, специфика), а расходов 6 разделов (функциональная группа, функциональная подгруппа, АБП, программа, подпрограмма, специфика)

Доход					2009-2010-2011-2012	2013-2014-2015-2016	2017-2018-2019-2020	Скорркт.	
К1	Кс	ПК	с	Дата окон	Наименование	Утвержденный бюджет	Отклонение	Уточненный	Скорркт.
1					Налоговые поступления	227 988 288,00	27 748 875,00	256 239 413,00	0,00
	01				Подходный налог	128 988 288,00	28 748 875,00	157 737 163,00	0,00
		2			Индивидуальный подоходный налог	128 988 288,00	28 748 875,00	157 737 163,00	0,00
			01		Индивидуальный подоходный налог с доходов, облагаемых у источника выплаты	128 988 288,00	28 748 875,00	157 737 163,00	0,00
			02		Индивидуальный подоходный налог с доходов, не облагаемых у источника выплаты	0,00	0,00	0,00	0,00
			04		Индивидуальный подоходный налог с доходов иностранных граждан, облагаемых у источника выплаты	0,00	0,00	0,00	0,00
			05		Индивидуальный подоходный налог с доходов иностранных граждан, не облагаемых у источника выплаты	0,00	0,00	0,00	0,00
	03				Социальный налог	86 470 000,00	7 980 000,00	94 450 000,00	0,00
	04				Налоги на собственность	27 482 475,00	0,00	27 482 475,00	0,00
	05				Внутренние налоги на товары, работы и услуги	8 933 000,00	0,00	8 933 000,00	0,00
	07				Прочие налоги	1 000,00	0,00	1 000,00	0,00
	08				Обязательные платежи, взимаемые за совершение юридически значимых действий и (или) выдачу документов уполномоченными на то государственными органами или должностными лицами	1 000 000,00	0,00	1 000 000,00	0,00
2					Неналоговые поступления	1 788 270,00	0,00	1 788 270,00	0,00
3					Поступления от продажи основного капитала	12 880 288,00	0,00	12 880 288,00	0,00
4					Поступления трансфертов	885 237 896,00	0,00	885 237 896,00	0,00
5					Погашение бюджетных кредитов	0,00	0,00	0,00	0,00
6					Поступления от продажи финансовых активов государства	0,00	0,00	0,00	0,00
7					Поступления займов	0,00	0,00	0,00	0,00
8					Используемые остатки бюджетных средств	0,00	0,00	0,00	0,00

Рисунок 3.5 – форма доходов

Справой стороны в основном всё так же но есть кнопка «загрузка данных» которая открывает форму для загрузок данных с Казначейства и Управления финансов.



Рисунок 3.6 –форма доходов

В Форме загрузок данных есть 3 основные панели.

Первая панель нужна для загрузки «Ежедневных поступлений» с Казначейства так же для удобства пользователя написана последняя дата загрузки, далее идет период в котором необходимо сделать анализ или удаление платежей.

Две другие панели служат для загрузки отчетов с Управления финансов.

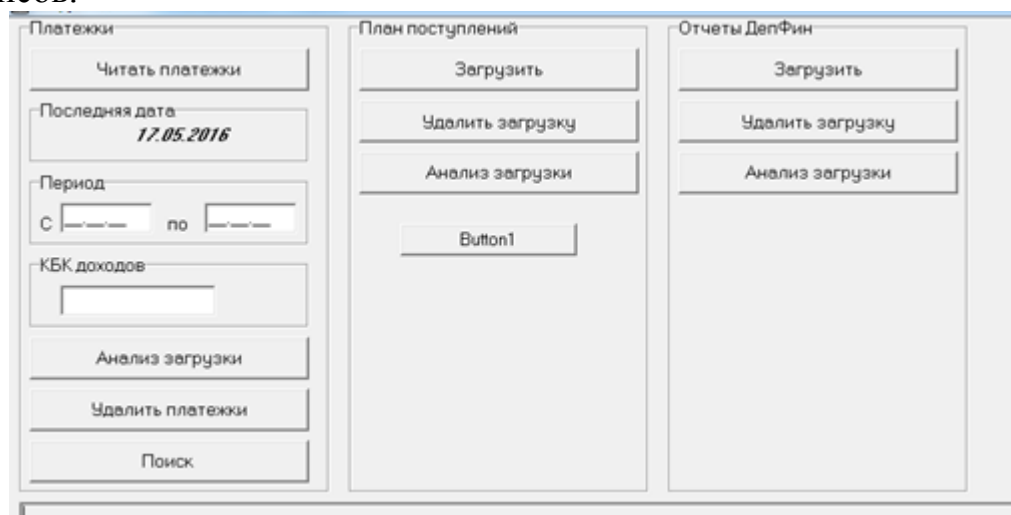


Рисунок 3.7 – Форма загрузки

Также основная задача данной программы является создание отчетов. В форме отчетов представлено более 30 разных видов отчетов. Так же реализовано отсортировка по отделам и каждый пользователь видит только те отчеты которые нужны для его отдела. Так как мы смотрим от администратора то мы можем видеть все отчеты.

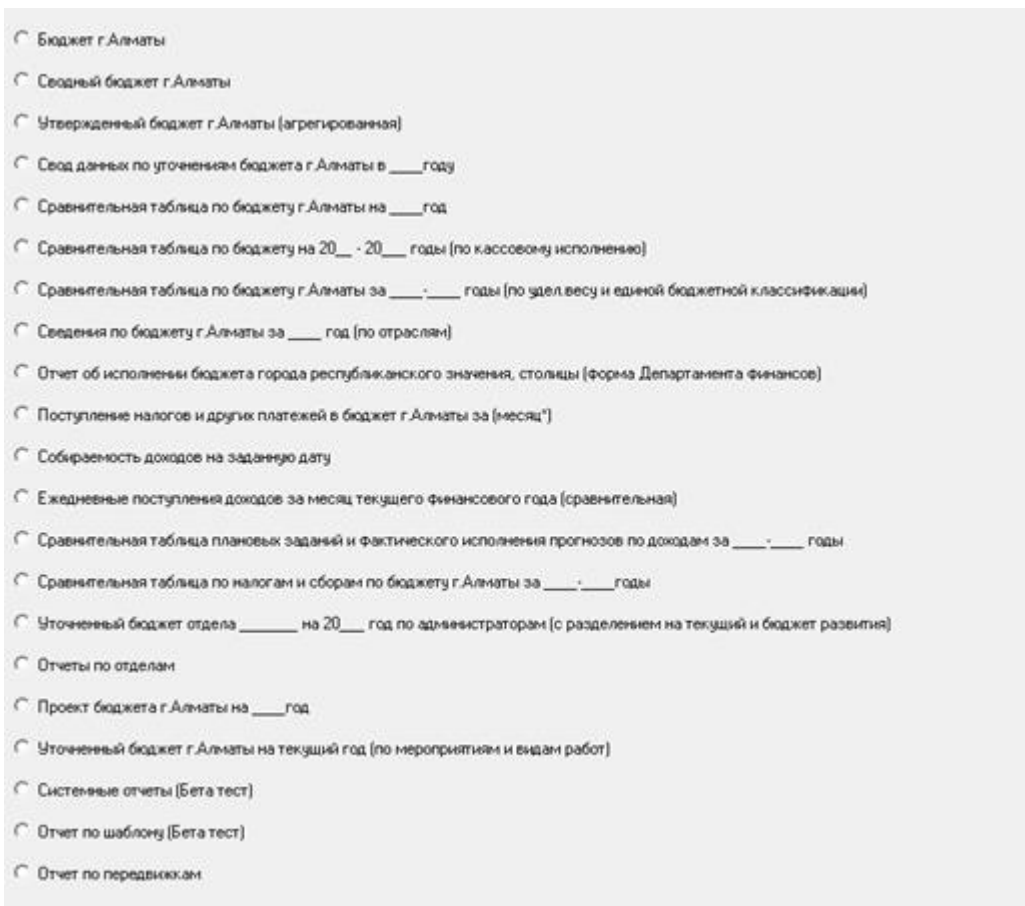


Рисунок 3.8 – Форма отчетов

Вначале мы должны выбрать группу отчетов, которые нам нужны далее мы видим форму с более точными отчетами

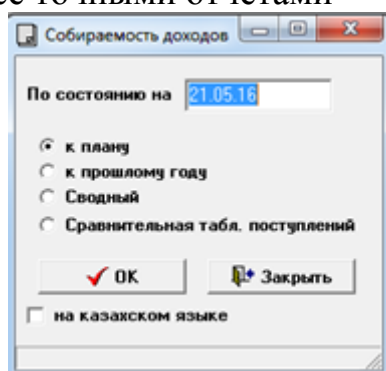


Рисунок 3.9 – Форма отчетов

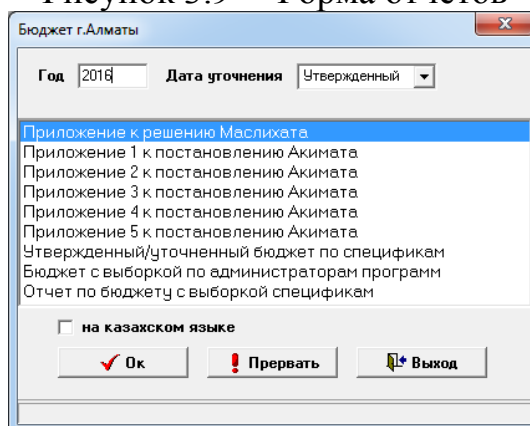


Рисунок 3.10 –форма отчетов

Так же администратору есть доступ к форме администрирования.

В ней администратор может создавать пользователей, редактировать отделы, администрировать БД, изменять справочники. Всё сделано на программном уровне, поэтому администратору не нужно специальных знаний для работы.

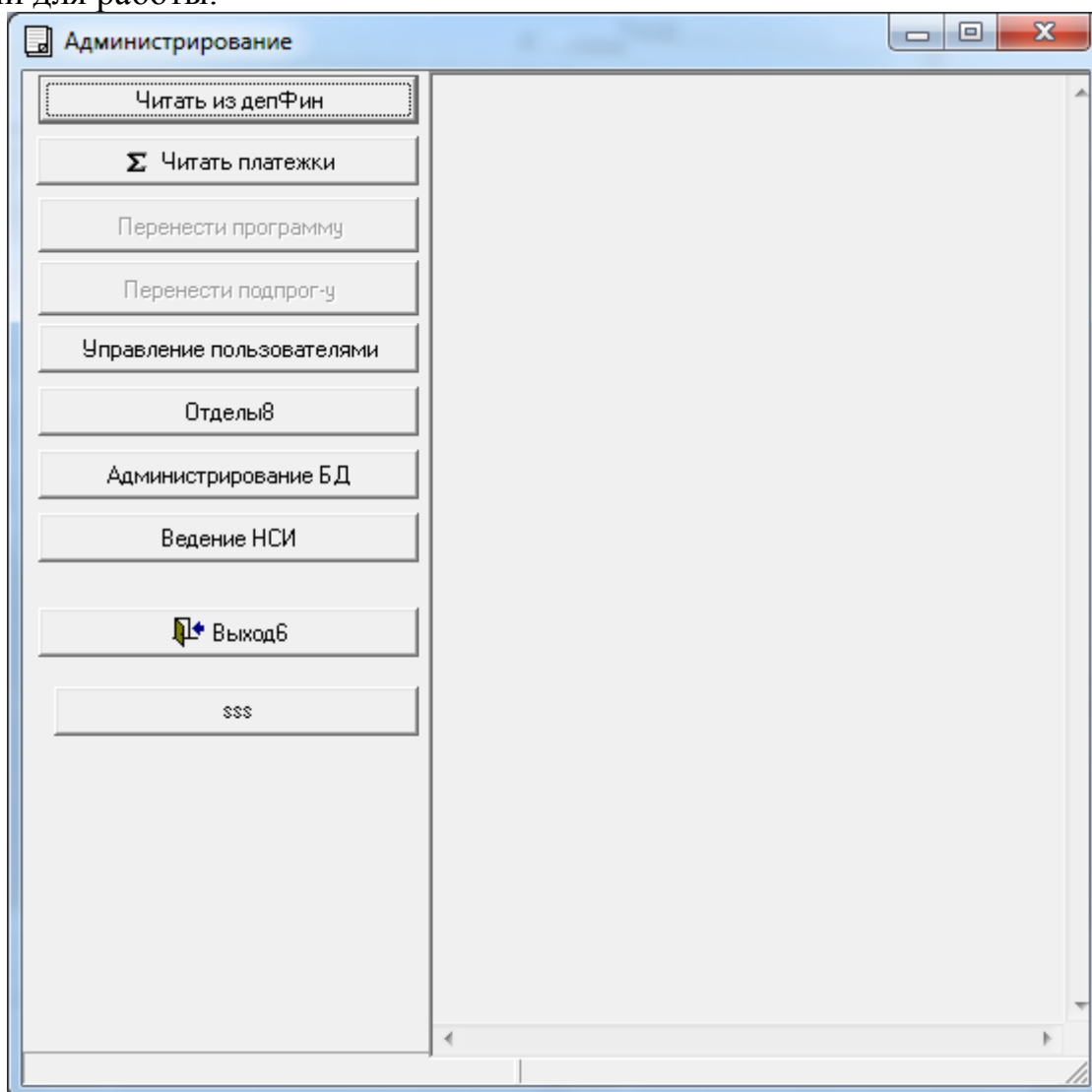


Рисунок 3.11- Форма администрирования

4 Техничко-экономическое обоснование

4.1 Описание работы и обоснование необходимости

Тема данного дипломного проекта – «АИС Бюджетного планирования».

Основной целью создания автоматизированной информационной системы «Бюджетное планирование г.Алматы» (далее – Системы) является автоматизация функций сотрудников Департамента экономики и бюджетного планирования для повышения оперативности при формировании проекта бюджета на следующий финансовый год.

4.2 Расчет затрат на разработку информационных технологий

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

$$C_{ni} = Z_{\text{фот}} + Z_{\text{сзи}} + A + Z_э + P_{ni} \quad (4.1)$$

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

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

A – амортизационные отчисления, тенге;

$Z_э$ – затраты на электроэнергию;

P_{ni} – накладные расходы, тенге. Размер фонда оплаты труда разработчиков (ЗФОТ) рассчитывается по формуле

Размер фонда оплаты труда разработчиков (ЗФОТ) рассчитывается по формуле

$$Z_{\text{фот}} = Z_{oi} + Z_{di} \quad (4.2)$$

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

Z_{di} – дополнительная заработная плата, тенге.

Затраты на оплату труда зависят от объема и трудоемкости разработки программного обеспечения. Общий объем (V_0) программного продукта определяется исходя из количества и объема функции, реализуемых программой

$$V_0 = \sum_n V_i \quad (4.3)$$

где V_i – объем отдельной функции ПО;

n – общее число функций.

$$V_0 = 42823 \text{ строк кода}$$

Общая трудоемкость проекта рассчитывается по формуле

$$T_0 = T_n * K_c * K_T * K_H \quad (4.4)$$

где K_c – коэффициент, учитывающий сложность ПО;

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

K_H – коэффициент, учитывающий степень новизны ПО;

T_n – нормативная трудоемкость.

Коэффициент сложности определяется на основе данных представленных в таблице 1 и составляет $K_c = 0,26$, т.к. в проекте присутствует более 3-х характеристик – функционирование системы, интерактивный доступ, обеспечение хранения, ведения и поиска данных в сложных структурах.

Таблица 4.1 – Дополнительные коэффициенты сложности ПО

Характеристика ПО	Значения K_c
1 Функционирование ПО в расширенной операционной среде (связь с другими ПО)	0,08
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	
4.1 2 характеристики	0,12
4.2 3 характеристики	0,18

4.3 свыше 3-х характеристик	0,26
-----------------------------	------

Поправочный коэффициент, учитывающий степень использования при разработке проекта стандартных модулей (K_T), определяется на основе данных представленных в таблице 3.2 и составляет 0,8.

Таблица 4.2 – Значения поправочного коэффициента, учитывающего использование стандартных модулей типовых программ и ПО (K_T)

Степень охвата реализуемых функций разрабатываемого ПО стандартными модулями, типовыми программами и ПО	Значения K_T
1 От 60% и выше	0,6
2 40% до 60%	0,7
3 20% до 40%	0,8
4 до 20%	0,9
5 Типовые программы и ПО не используемык реализации функций разрабатываемого ПО	1,0

Поправочный коэффициент, учитывающий новизну разрабатываемого проекта (K_H) определяется на основе данных, представленных в таблице 3.3 и составляет 0,7.

Таблица 4.3 – Поправочные коэффициенты, учитывающие новизну ПО (K_H)

Категория новизны	Степень новизны	Использование		Значение K_H
		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 ПК и ОС	-	-	0,7

Базой для определения нормативной трудоемкости являются, укрупненные нормы времени на разработку проекта в зависимости от уточненного объема ПО и группы сложности.

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

Учитывая данные из Приложения В: для 1–ой категории сложности ПО

$$T_n = 1161.$$

Общий объем трудоемкости исходя из формулы (4.4)

$$T_0 = 1161 * 0,8 * 0,7 * 0,26 = 169,04 \text{ чел.-час.}$$

На основе трудоемкости определяются плановое число разработчиков ($Ч_p$) и плановые сроки, необходимые для реализации проекта в целом (T_p). При этом могут решаться следующие задачи:

- расчет числа исполнителей при заданных сроках разработки проекта;
- определение сроков разработки проекта при заданной численности исполнителей.

Численность исполнителей проекта ($Ч_p$) рассчитывается по формуле

$$Ч = T_0 / (T_p * \Phi_{эф}) \quad (4.5)$$

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

$$T_p = T_0 / (Ч_p * \Phi_{эф}) \quad (4.6)$$

где $Ч_p$ – плановое число разработчиков.

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

$$\Phi_{эф} = D_r - D_{п} - D_{в} - D_0 \quad (4.7)$$

где D_r – количество дней в году;

$D_{п}$ – количество праздничных дней в году;

$D_{в}$ – количество выходных дней в году;

D_0 – количество дней отпуска.

Т.к., в соответствии с производственным календарем на 2016 год:

$$D_r = 366;$$

$$D_{п} = 17;$$

$$D_{в} = 105;$$

$D_0 = 10$, эффективный фонд времени одного работника составит

$$\Phi_{эф} = 366 - 17 - 105 - 10 = 234 \text{ дня}$$

Плановое число разработчиков $Ч_p = 1$, следовательно, по формуле (4.6)

$$T_p = 169,04 / (1 * 234) = 0,72 \text{ года}$$

Основная заработная плата исполнителей на конкретное ПО рассчитывается по формуле

$$Z_{oi} = \sum_n T_{ci} * T_c * K \quad (4.8)$$

где n – количество исполнителей, занятых разработкой конкретного ПО;

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

Φ_{π} – плановый фонд рабочего времени i–го исполнителя (дней), число рабочих дней в месяц;

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

K – коэффициент премирования, составляет 1,38.

Так как на разработку уйдет ≈ 7 месяцев, а число рабочих дней в каждом месяце разное, возьмем среднее значение $\Phi_{\pi}=21$ раб.дн.

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 3.4).

Таблица 4.4 - Сведения по работникам, задействованным в проекте

Специалист-исполнитель	Количество, человек	Заработная плата в месяц, тенге
Инженер-программист	1	150 000,00
Итого	1	150 000,00

Часовая тарифная ставка рассчитывается путем деления месячной тарифной ставки, установленную при 40–часовой недельной норме рабочего времени и общего фонда времени (Φ_p)

$$T_c = \frac{T_m}{\Phi_p} \quad (4.9)$$

где T_c – часовая тарифная ставка (тыс.тенге);

T_m – месячная тарифная ставка (тыс.тенге).

Общий фонд времени

$$\Phi_p = T_c * \Phi_{\pi} \quad (4.10)$$

Таким образом

$$\Phi_p = 8 * 21 = 168 \text{ часа}$$

Тарифная ставка разработчика проекта

$$T_{ci} = \frac{150000}{168} = 893 \text{ тенге в час}$$

Результаты расчета основной заработной платы представлены в виде таблице 3.5.

Таблица 4.5 – Сводные результаты расчета затрат основной заработной платы

Наименование содержания работ	Исполнитель	Трудоёмкость норма-час	Заработная плата за час работы, т/час	Сумма заработной платы, тенге

ТЗ	Программист	244	893	218 094
Моделирование	Программист	567	893	506 471
Программирование	Программист	933	893	832 983
Тестирование	Программист	202	893	179 959
Внедрение	Программист	156	893	139 145
Итого		2102		1 876 651

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

$$З_{дi} = З_{oi} * N_{д} / 100 \quad (4.12)$$

где $N_{д}$ – коэффициент дополнительной заработной платы разработчиков

$$З_{дi} = 1876651 * 0,10 = 187665,1 \text{ тенге}$$

Таким образом, суммарный фонд оплаты труда составит:

$$\Phi OT = 1876651 + 187665,1 = 2064316 \text{ тенге}$$

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

$$З_{сzi} = (\Phi OT - \text{ПО}) * 11\% \quad (4.13)$$

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

$$\text{ПО} = \Phi OT * 10\% \quad (4.14)$$

Таким образом

$$\text{ПО} = 2064316 * 0,1 = 206431,6 \text{ тенге}$$

$$З_{сzi} = (2064316 - 206431,6) * 0,11 = 204367,284 \text{ тенге}$$

4.3 Оборудование, используемое в работе

Оборудование, используемое при разработке портала, представлено в таблице 4.2.

Т а б л и ц а 4.6 – Оборудование, необходимого для разработки портала

Наименование	Характеристики	Кол-во	Стоимость без НДС, тенге
Сервер	Asus TS300-E8-PS4 Tower (S1150 Xeon, iC224, 4 DDR3 ECC, 4xGLAN, VGA, 6SATA)	1	232 174

4.4 Программное обеспечение, используемое в работе

ПО, используемое в работе, представлено в таблице 4.3.

Т а б л и ц а 4.7 – Перечень программного обеспечения, используемого при разработке портала.

Программное обеспечение	Стоимость без НДС, тенге
Microsoft Office Professional Plus	66 000
Sybase 12.5 Express Editon	1 849 717
Итого:	1 915 717

4.5 Расчет амортизационных отчислений

Амортизационные отчисления рассчитываются по формуле:

$$A_i = \frac{N_A * C_{пер} * N}{100 * 12 * n} \quad (4.15)$$

где N_A – норма амортизации;

$C_{пер}$ – первоначальная стоимость оборудования;

N – количество дней на выполнение работ;

n – количество дней в рабочем месяце.

Норма амортизации N_A на компьютерную технику и на программное обеспечение составляет 40% от всей стоимости, а на лицензию – 15%.

Амортизационные отчисления по используемому оборудованию и программному обеспечению составят:

$$A_1 = \frac{40 * 232\,174 * 262}{100 * 365} = 66\,866 \text{ тенге};$$

$$A_3 = \frac{40 * 66\,000 * 262}{100 * 365} = 18\,950 \text{ тенге};$$

$$A_4 = \frac{15 * 1\,849\,717 * 262}{100 * 365} = 199\,161 \text{ тенге};$$

$$A = 66\,866 + 18\,950 + 199\,161 = 284\,977 \text{ тенге}.$$

4.6 Расчет затрат на электроэнергию

Так как, в процессе производства используется электрооборудование необходимо рассчитать затраты на электроэнергию. Затраты на электроэнергию для производственных нужд включают в себя расходы: электроэнергии на оборудование и на дополнительные нужды.

$$\Theta = \Theta_{эл.эн.обор} + \Theta_{доп.нуж} \quad (4.16)$$

где $\Theta_{эл.эн.обор}$ – затраты на электроэнергию оборудования;

$\Theta_{доп.нуж}$ – затраты электроэнергии на дополнительные нужды.

Расходы электроэнергии на оборудование рассчитываются по формуле:

$$\Theta_{эл.эн.обор} = W * T * S * K_{исп} \quad (4.17)$$

где: W – потребляемая мощность, Вт;

T – количество часов работы оборудования;

S – стоимость киловатт-часа электроэнергии (1кВтч = 16,02 тг без НДС);

$K_{исп}$ = коэффициент использования ($K_{исп} = 0,9-1$).

$W = 550 \text{ Вт} = 0,55 \text{ кВт}$ (Мощность сервера);

Затраты на электроэнергию основного оборудования согласно формуле 4.11 составляет:

$$Z_{эл.эн.обор} = 0,55 * (262*8) * 16,02 * 0,9 = 16\ 621 \text{ тенге.}$$

Затраты на дополнительные нужды берутся по укрупненному показателю в размере 5% от затрат на оборудование:

$$Z_{доп.нуж} = 0,05 * Z_{эл.эн.обор} \quad (4.18)$$

$$Z_{доп.нуж} = 0,05 * 16\ 621 = 831 \text{ тенге.}$$

Суммарные затраты на электроэнергию составляют:

$$\mathcal{E} = 16\ 621 + 831 = 17\ 452 \text{ тенге.}$$

Затраты по статье «Накладные расходы» ($P_{ни}$), рассчитывается по нормативу ($H_{рн}$) в процентном отношении к основной заработной плате исполнителей. Норматив устанавливается в целом по организации

$$P_{ни} = Z_{oi} * H_{рн} \quad (4.19)$$

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

$H_{рн}$ – норматив накладных расходов в целом по организации 40%.

Таким образом, накладные расходы составят

$$P_{ни} = 1\ 876\ 651 * 0,4 = 750\ 660,4 \text{ тенге}$$

Полные затраты на разработку АИС Бюджетное планирование формула (4.1)

$$C_{ни} = 207033 + 204367 + 284\ 977 + 17\ 452 + 750\ 660 = 3\ 134\ 108 \text{ тенге}$$

Таблица 4.8 – Затраты на разработку АИС Бюджетное планирование

Затраты на разработку	Условное обозначение	Значение, тенге	В процентах от общей суммы
Фонд оплаты труда	$Z_{фот}$	1 876 651	60
Социальный налог	$Z_{сзи}$	204367	7
Амортизационные отчисления	A	284 977	9
Затраты на электроэнергию	$Z_э$	17 452	1
Накладные расходы	$P_{ни}$	750 660	24
Итого:		3 134 108	100

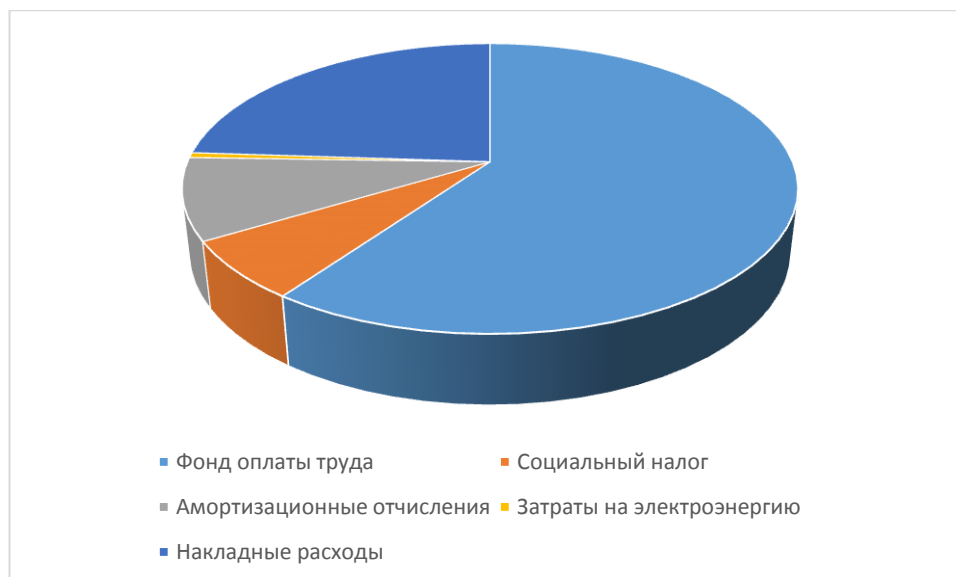


Рисунок 4.1 – Структура затрат на разработку АИС Бюджетное планирование

4.7 Расчет цены программного продукта

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

$$Ц_{ПП} = Z_{РПР} + П_{П} + НДС \quad (4.20)$$

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

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

$П_{П}$ – планируемая прибыль, тенге;

НДС – налог на добавленную стоимость, тенге. Планируемая прибыль составляет (20%) от себестоимости разработки

$$П_{П} = 3\,134\,108 * 0,2 = 626\,821,6 \text{ тенге}$$

НДС, начисленный на $П_{П}$, определяется следующим образом

$$НДС = (Z_{РПР} + П_{П}) * k_{НДС} \quad (4.21)$$

где $k_{НДС}$ – ставка налога на добавленную стоимость. Подставив данные в формуле (3.21) получаем

$$НДС = (3\,134\,108 + 626\,821,6) * 0,12 = 451\,311,552 \text{ тенге}$$

Подставив данные в формуле (3.20) получаем

$$Ц_{ПП} = 3\,134\,108 + 626\,821,6 + 451\,311,552 = 4\,212\,241,152 \text{ тенге}$$

Вывод

В результате экономического расчета стоимость разработки информационной системы «Бюджетного планирования» составила 3 134 108 тенге. Основной статьей расходов является заработная плата, которая составляет 60% от всех затрат. Данная система повышает оперативность при формировании проекта бюджета на следующий финансовый год, так же облегчает составление отчетов для акимата, и является хранением данных поступающие с Казначейства и Управления финансов. Этим она

оптимизирует работу людей в управлении. Цена реализации программного продукта составила 4 212 241,152 тенге. Стоимость данной системы относится к дорогим продуктам.

5 Безопасность жизнедеятельности

5.1 Анализ условий рабочего помещения

Главной целью данной работы является разработка программного обеспечения «Бюджетное планирование». Данный программный продукт является ПО для учета доходов и расходов.

Технический персонал состоит из нескольких сотрудников – Программист-разработчик, администратор базы данных.

Работа сотрудников связана с компьютером, ввиду этого появляются дополнительные группы вредных факторов, что вредит и ухудшает производительности их труда.

К таким факторам можно отнести:

- 1) Пожароопасность серверной комнаты
- 2) Ненормативная температура комнаты
- 3) Вредное излучение от персонального компьютера

Вследствие того, что серверы работают 24 часа в сутки возникает повышенный фактор возгорания из-за разного рода замыканий или скачков напряжения, так же из-за несоблюдения ППБ при проектировании серверной комнаты.

Согласно СП 5.13130.2009

- 1) Предел огнестойкости серверной должен быть следующим: перегородки - не менее EI 45, стены и перекрытия - не менее REI 45. Т.е. в условиях пожара помещение должно оставаться герметичным в течение 45 минут, препятствуя дальнейшему распространению огня.
- 2) Помещение серверной должно быть отдельным помещением, функционально не совмещенным с другими помещениями. К примеру, не допускается в помещении серверной организовывать мини-склад IT-оборудования или канцелярских товаров.

Для более безопасного в серверную комнату необходимо поставить автоматическую установку пожаротушения. Что касается типа установки пожаротушения, то здесь существует несколько вариантов

- 1) водяное пожаротушение
- 2) порошковое пожаротушение
- 3) аэрозольное пожаротушение
- 4) пожаротушение тонко-распыленной водой
- 5) газовое пожаротушение

Для серверной комнаты единственным разумным вариантом является установка газового пожаротушения, потому что остальные нанесут фатальный урон оборудованию. Не стоит забывать и о системах автоматического обнаружения пожара. Они должны стоять не только на потолке, но и у серверов и источников питания, так как из-за выделенной системы охлаждения для серверов дым будет идти по потоку воздуха и может миновать потолочный датчик.

При работе в офисе при ненормативной температуре внешней среды, так же вызывают ухудшение состояния человека и уменьшение производительности.

- 1) Холодный период года:
 - a. температура 19 – 24°С;
 - b. относительная влажность 40 – 70 %;
 - c. скорость движение воздуха относительная 0,2 м/с.
- 2) Тёплый период года:
 - a. температура 25 – 28°С;
 - b. относительная влажность 50 – 70;
 - c. скорость движение воздуха относительная 0,12 м/с.

Для нормального психологического состояния человека, который будет пользоваться ПО, необходимо оптимизированный продукт без зависаний системы. В фактор зависаний влияет не сколько факторов это персональный компьютер на котором идет работа, оптимизированный код в самом ПО и нормальная работа сервера. Чтобы решить эту проблему необходимо:

- 1) Компьютеры с рекомендованными требованиями
- 2) Сервер с рекомендованным требованиям
- 3) Своевременное обновление программного обеспечения
- 4) Выделенная линия связи сервера с персональными компьютерами на которых поставлено программное обеспечение

Также для сервера нужно выделенное охлаждение в следствии постоянного включенного состояния.

Помещение представляет собой две комнаты: серверная и рабочая комната.

5.2 Расчет системы кондиционирования кабинета.

Кондиционирование воздуха выполняет задачи вентиляции и создает благоприятные микроклиматы. Кондиционирование выполняется в соответствиях с главой СНиП 11-33-75 "Отопление, вентиляция и кондиционирование воздуха".

Таблица 4.1. Оптимальные нормы температуры, относительной влажности и скорости движения в обслуживаемой зоне жилых, общественных и административно-бытовых помещений

Таблица 5.1 – Температура воздуха в помещении.

Период года	Категория работ	Температура воздуха,	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	Легкая – 1а	22-24	40-60	0,1
	Легкая – 1б	21-23	40-60	0,1
Теплый	Легкая – 1а	23-25	40-60	0,1
	Легкая – 1б	22-24	40-60	0,2

Найдем количество приточного воздуха для более точного выбора кондиционера. Для этого используем формулу:

$$L_{np} = \frac{Q_{изб}}{c * P_{np} * (t_{выт} - t_{пр})} \quad (5.1)$$

Где: $Q_{изб}$ – избыточное выделение явной теплоты, кДж/ч;

c – удельная теплоемкость воздуха при постоянном давлении, равная $c = 1$ кДж/кг* $^{\circ}$ С;

P_{np} – плотность поступающего в помещение воздуха, 1,2кг/м 3

$t_{выт}$ – температура удаляемого из помещения воздуха за пределы рабочего или обслуживаемой зоны, $^{\circ}$ С;

$t_{пр}$ – температура приточного воздуха, $^{\circ}$ С;

Формула для нахождения температуры удаляемого из помещения воздуха за пределы рабочего или обслуживаемой зоны :

$$t_{выт} = t_{рз} + \Delta t * (h_{вп} - z) \quad (5.2)$$

Где:

$t_{рз}$ – температура в рабочей зоне, которые не должна превышать допустимую по нормам ($t_{рз} \leq t_{доп}$), $^{\circ}$ С;

$h_{вп}$ – расстояние от пола до центра вытяжных проемов (кондиционера), м;

z – высота рабочей зоны, м

Производим расчёт для теплого периода года, возьмем $t_{рз} = 23^{\circ}$ С;

Внутренняя часть кондиционера расположена на высоте $h_{вп} = 2$ м

$$t_{выт} = 23 + 1,2 * (2 - 1) = 24,2^{\circ}$$

Температура приточного воздуха $t_{пр}$ при наличии избытка явной теплоты должна быть на 5 – 7 $^{\circ}$ С ниже температуры воздуха в рабочей зоне
 $t_{пр} = 24,2 - 6 = 18,2^{\circ}$ С

Величину избыточного выделения явной теплоты $Q_{изб}$ находят на основании баланса теплоты в помещении по формуле:

$$Q_{изб} = \Sigma Q - \Sigma Q_{ух} \quad (5.3)$$

где ΣQ – суммарное количество поступающей в помещение явной теплоты.

$\Sigma Q_{ух}$ – суммарное количество уходящей из помещения теплоты (за счет тепло потерь ограждениями, нагрева поступающего в помещение воздуха и т.п.)

В помещении источниками избыточного тепла являются человек, источник искусственного освещения, тепловыделения электронного оборудования, излучение солнца. Так как тепловыделения электронного оборудования незначительна им можно пренебречь.

Для расчета тепловыделения от искусственного света Q_1 предположим что вся затрачиваемая энергия преобразуется в тепло.

$$Q_1 = N * n \quad (5.4)$$

Где N – расходуемая мощность светильника, Вт

n – количество светильников.

$$Q_1 = 100 * 5 = 500 \text{ Вт}$$

Тепловыделения от людей Q_2 определяется по формуле:

$$Q_2 = n * q \quad (5.5)$$

Где n – число работающих,

q – количество тепла, выделяемое одним человеком, представлено в таблице 5.2

Таблица 5.2. Количество тепла, выделяемое одним человеком

Категория работ	Тепло, Вт			
	Полное		Явное	
	При 100°С	При 350°С	При 100°С	При 350°С
Легкая	180°С	145°С	150°С	5°С

$$Q_2 = 2 * 145 = 290 \text{ Вт}$$

Количество тепла от солнечной радиации определяется формулой:

$$Q_{\text{ост. рад}} = F_{\text{ост}} * q_{\text{ост}} * A_{\text{ост}} \quad (5.6)$$

где $F_{\text{ост}}$ – площадь поверхности и покрытия, м^2 ;

$q_{\text{ост}}$ – теплопоступления через 1 м^2 поверхности остекления и поверхности покрытия, при коэффициенте теплопередачи, равном $1 \text{ Вт/м}^2 * ^\circ\text{С}$

$A_{\text{ост}}$ – коэффициент остекления;

$$F_{\text{ост}} = 2 * 4 = 8 \text{ м}^2$$

Окно рабочего помещения направлено на юг поэтому $q_{\text{ост}}$ равен $150 \text{ Вт/м}^2 * ^\circ\text{С}$

$$A_{\text{ост}} = 0,5$$

$$Q_{\text{ост. рад}} = 8 * 150 * 0,5 = 600 \text{ Вт}$$

Среднее значение теплопоступления для покрытия с учетом географической широты примем равным $Q_{\text{п.рад}} = 8 \text{ Вт}$.

Потери тепла из помещения $Q_{\text{ух}}$, кВт через стены двери, окна оценивают ориентировочно по формуле:

$$Q_{\text{ух}} = \frac{\lambda * S * (t_{\text{внут}} - t_{\text{нар}})}{\delta} \quad (5.7)$$

где λ – теплопроводность стен, $\text{Вт/м}^2 * ^\circ\text{С}$.

S – площадь, м^2

δ – толщина стен, м.

Стены рабочего помещения изготовлены из кирпичей М350, теплопроводность которого равна $0,7 \text{ Вт/м}^2 \cdot ^\circ\text{С}$. Толщина стен $\delta = 0,3 \text{ м}$.

$$Q_{yx} = 0,7 \cdot 50 \cdot (22 - 16) / 0,3 = 700 \text{ Вт.}$$

Вычислим суммарное количество поступающей в помещение явной теплоты.

$$\sum Q = Q_1 + Q_2 + Q_{\text{ост.рад}} + Q_{\text{пр.п}} = 500 + 290 + 600 + 8 = 1398 \text{ Вт} \quad (5.8)$$

Вычислим количество приточного воздуха;

$$L_{\text{пр}} = 1398 / (1 \cdot 1,2 \cdot (24,2 - 18,2)) = 194 \text{ м}^3/\text{ч}$$

Выбираем кондиционер с нижней подачей Elenberg CSH-07J с максимальным расходом воздуха $L=300 \text{ м}^3/\text{ч}$, что является оптимальным для обеспечения благоприятного микроклимата.

Таблица 5.3 Характеристика Elenberg CSH-07J

Площадь помещения	21 м ²
Мощность охлаждения	1000 Вт
Потребляемая мощность при охлаждении	750 Вт
Мощность обогрева	1000 Вт
Потребляемая мощность при обогреве	750 Вт



Рисунок 4.1 Фото кондиционера Elenberg CSH-07J

5.3 Расчет искусственного освещения точечным методом

Для расчета искусственного света воспользуемся точечным методом. Для этого посчитаем что лампы в комнате это точечный излучатель.

Чтобы вычислить нам понадобится:

Нормативная освещенность – 400 лк.

Высота подвеса светильников над освещаемой поверхностью $H=2,45 \text{ м}$,

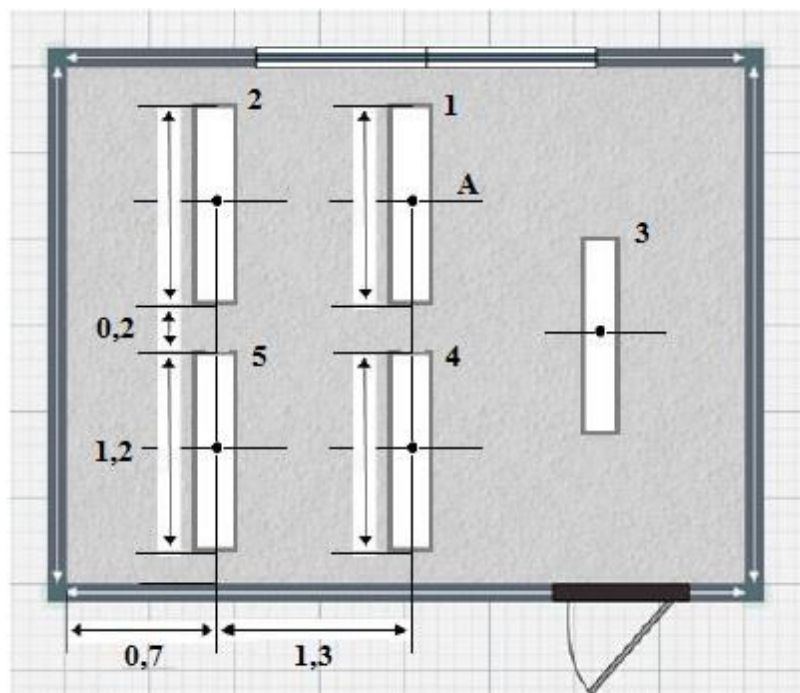


Рисунок 4.2 – Схема освещенности

Для точки А определяем суммарную условную освещенность всех светильников следующим образом.

Определяем проекцию расстояния на потолок от точки А до светильников. В дальнейшем определяем косинус угла между потолком и прямой.

$$\alpha = \arctg\left(\frac{d_i}{H}\right) \quad (5.9)$$

d_i – расстояние от контрольной точки до i -ой лампы.

H – высота подвеса

$$e_i = \frac{I_\alpha \cdot \cos^3(\alpha) \cdot \mu}{k \cdot H^2} \quad (5.10)$$

I_α - сила света в направлении от источника на заданную точку рабочей поверхности (определяют по кривым силы света или по таблицам выбранного типа светильника),

α - угол между нормалью к рабочей поверхности и направлением силы света к расчетной точке,

μ - коэффициент, учитывающий действие удаленных от расчетной точки светильников и отраженного светового потока от стен, потолка, пола, оборудования, падающего на рабочую поверхность в расчетной точке (принимают в пределах $\mu = 1,05 \dots 1,2$),

k - коэффициент запаса,

H - высота подвеса светильника над рабочей поверхностью.

Так как точка А находится под лампой 1 угол $\alpha=0$. Определяем I_α . По кривой силы света светильников ЛД при условной лампе со световым потоком $\Phi_{л} = 1000$ лм. $I_\alpha = 365$ кд.

Световой поток установленной в светильнике лампы ДРЛ мощностью 100 Вт равен 4750 лм. Конечный результат $I_{\alpha} = 365 \cdot (4750/1000) = 1735$ кд.

$$e_i = \frac{I_{\alpha} \cdot \cos^3(\alpha) \cdot \mu}{k \cdot H^2} = \frac{1735 \cdot 1 \cdot 1.05}{1.5 \cdot 2,25^2} \approx 240 \text{лк}$$

Далее рассчитываем относительную освещенность от светильника 3

$$d_3 = \sqrt{(0.2 + 0.6)^2 + 1.3^2} = 1.53$$

$$\alpha = \arctg\left(\frac{1.53}{2.25}\right) = 35$$

Определяем I_{α} . По кривой силы света светильников ЛД при условной лампе со световым потоком $\Phi_{\text{л}} = 1000$ лм $I_{\alpha} = 164,25$ кд.

$$I_{\alpha} = 164,25 \cdot (4750/1000) = 780,1875 \text{ кд.}$$

$$e_i = \frac{I_{\alpha} \cdot \cos^3(\alpha) \cdot \mu}{k \cdot H^2} = \frac{780,1875 \cdot 0.5498 \cdot 1.05}{1.5 \cdot 2,25^2} \approx 59 \text{лк}$$

Далее рассчитываем относительную освещенность от светильника 2

$$\alpha = \arctg\left(\frac{1.3}{2.25}\right) = 30$$

Определяем I_{α} . По кривой силы света светильников ЛД при условной лампе со световым потоком $\Phi_{\text{л}} = 1000$ лм $I_{\alpha} = 291$ кд.

$$I_{\alpha} = 291 \cdot (4750/1000) = 1382,25 \text{ кд.}$$

$$e_i = \frac{I_{\alpha} \cdot \cos^3(\alpha) \cdot \mu}{k \cdot H^2} = \frac{1382,25 \cdot 0.65 \cdot 1.05}{1.5 \cdot 2,25^2} \approx 124 \text{лк}$$

Далее рассчитываем относительную освещенность от светильника 4

$$\alpha = \arctg\left(\frac{0,8}{2.25}\right) = 20$$

Определяем I_{α} . По кривой силы света светильников ЛД при условной лампе со световым потоком $\Phi_{\text{л}} = 1000$ лм $I_{\alpha} = 316$ кд.

$$I_{\alpha} = 316 \cdot (4750/1000) = 1501 \text{ кд.}$$

$$e_i = \frac{I_{\alpha} \cdot \cos^3(\alpha) \cdot \mu}{k \cdot H^2} = \frac{1501 \cdot 0,8297 \cdot 1.05}{1.5 \cdot 2,25^2} \approx 172,2 \text{лк}$$

Так как светильник номер 5 находится сильно отдаленно от рабочего места мы его не берем в расчет. Далее мы высчитываем суммарную освещенность $\sum e_i = 240 + 59 + 124 + 172,2 = 595,2 \text{лк}$

Так как нормативная освещенность равна 400 лк, а по нашим расчетам суммарная освещенность в офисе равна 595,2лк, сделан вывод что в офисе что искусственного освещения предостаточно для комфортной работы.

Заключение

В данной дипломной работе мною было спроектировано и разработана информационная система «Бюджетное планирование» которая облегчает Управлению экономики и бюджетного планирования созданию бюджета на следующий год, так же решает проблемы по хранению данных. В качестве СУБД было использовано Sybase ASE и она показала себя с лучшей стороны своей защищенностью и быстродействию даже перед большой базой.

Для достижения основных целей и задач были проделаны следующие работы

- изучены требования заказчика;
- спроектирована база данных;
- создана система для работы с базой данных;

Дизайн был сделан максимально понятным и полнофункциональным для эффективного использования заказчиком.

На основе расчетов экономической части было просчитано что данная система стоит 3 134 108 тенге. Где основная часть 60% является заработной платой. Цена реализации системы составила 4 212 241,152 тенге.

Итоговая цена разработанного продукта является более чем выгодным, причем обладает многими функциональными возможностями.

Список литературы

Г.М. Гукасян. Экономика от А до Я: Тематический справочник. – М.: ИНФРА-М, 2009. – 480 с.

Экономика промышленного предприятия: учебник / И.Н. Иванов. – М.: ИНФРА-М, 2011. – 395 с.

Ю.М. Остапенко. Экономика труда: учебное пособие. – М.: ИЦ РИОР, 2010. – 160 с.

А.И. Рофе. Экономика труда: учебник. – М.: КиноРус, 2010. – 400 с.

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

СП 5.13130.2009

Белов С.В., Ильницкая А.В., Козьяков А.Ф. Безопасность жизнедеятельности. Учебник для вузов. – М.: Высшая школа, 2005. – 448 с.

СНиП РК 4.02–42–2006 Отопление, вентиляция и кондиционирование. – Астана: Издательство стандартов, 2007.

СНиП РК 2.04-05-2002 Естественное и искусственное освещение. Государственные нормативы в области архитектуры, градостроительства и строительства. - Издательство стандартов Астана, 2010 г.

Абдимуратов Ж.С, Маманбаев С.Е. Безопасность жизнедеятельности. Методические указания к выполнению раздела «Расчет производственного освещения» в выпускных работах для всех специальностей. Бакалавриат. - Алматы: АУЭС, 2009.

Приложение А

Листениг программы

```
unit StartUnit;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, Buttons, unitDM, UnitMyFunctions, IniFiles, ShellApi,  
jpeg, ExtCtrls;
```

```
type
```

```
TfrmStart = class(TForm)  
  btn1Enter: TBitBtn;  
  btnReport: TBitBtn;  
  btnAdm: TBitBtn;  
  btn2Enter: TBitBtn;  
  btn3Enter: TBitBtn;  
  btn5Enter: TBitBtn;  
  BitBtn8: TBitBtn;  
  Image1: TImage;  
  btn4Enter: TBitBtn;  
  BtnLoadNew: TBitBtn;  
  readXml: TBitBtn;  
  btnReports: TBitBtn;  
  btn6Enter: TBitBtn;  
  btnDemand: TButton;  
  Button1: TButton;  
  btn8Enter: TButton;  
  BitBtnlang: TBitBtn;  
  btn9Enter: TButton;  
  procedure BitBtn8Click(Sender: TObject);  
  procedure btn1EnterClick(Sender: TObject);  
  procedure btnReportClick(Sender: TObject);  
  procedure FormCreate(Sender: TObject);  
  procedure btnAdmClick(Sender: TObject);  
  procedure FormClose(Sender: TObject; var Action: TCloseAction);  
  procedure btn2EnterClick(Sender: TObject);  
  procedure btn3EnterClick(Sender: TObject);  
  procedure btn5EnterClick(Sender: TObject);  
  procedure btn4EnterClick(Sender: TObject);  
  procedure FormShow(Sender: TObject);  
  procedure BtnLoadNewClick(Sender: TObject);  
  procedure readXmlClick(Sender: TObject);
```

```

procedure btnReportsClick(Sender: TObject);
procedure btn6EnterClick(Sender: TObject);
procedure btnDemandClick(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure btn9EnterClick(Sender: TObject);
procedure btn8EnterClick(Sender: TObject);
procedure addLog(Userlog, dast, dep, Where_, Wat_: string);
procedure BitBtn8Exit(Sender: TObject);
function IniGetStringValue( IniSection: string;StringName:
string;DefaultString: string): string;
procedure LoadLang;
procedure BitBtnlangClick(Sender: TObject);
private
  { Private declarations }
public
  from_:string;
  langIni: TIniFile;
  sPath: String[160];
  // lang:String[3];

end;

var
  frmStart: TfrmStart;
  gExpensesTable, gActionTable, gProcedureForShow, gProcedureForDelShow,
gProcedureRestore,
  gProcedureUpd :string;

implementation

uses MainUnit, RepUnit1, UnitScreen, UnitAdditional,
  UnitEduCultur, UnitSocial, UnitGosApparat, UnitRealSector,
  UnitAuthenticate, UnitReadXml, UnitConsReport, UnitSvod, UnitMain,
  RepSelect, UnitSelect, Math, LangUnit, UnitRealSector1;

{$R *.dfm}

function TfrmStart.IniGetStringValue(
  IniSection: string;
  StringName: string;
  DefaultString: string): string;

begin

```

```

try
  Result :=
    langIni.ReadString(
      IniSection,
      StringName,
      DefaultString);
  if Result = '' then
    Result := 'нет';
finally

end;
end;

procedure TfrmStart.LoadLang;
begin
frmStart.Caption:=IniGetStringValue('StartUnit','FrmCaption','err');
btn1Enter.Caption:=IniGetStringValue('StartUnit','btn1Caption','err');
btn2Enter.Caption:=IniGetStringValue('StartUnit','btn2Caption','err');
btn3Enter.Caption:=IniGetStringValue('StartUnit','btn3Caption','err');
btn4Enter.Caption:=IniGetStringValue('StartUnit','btn4Caption','err');
btn5Enter.Caption:=IniGetStringValue('StartUnit','btn5Caption','err');
btn6Enter.Caption:=IniGetStringValue('StartUnit','btn6Caption','err');
btn8Enter.Caption:=IniGetStringValue('StartUnit','btn8Caption','err');
BitBtnlang.Caption:=IniGetStringValue('StartUnit','BitBtnlangCaption','err');

btnReport.Caption:=IniGetStringValue('StartUnit','btnReportCaption','err');
btnDemand.Caption:=IniGetStringValue('StartUnit','btnDemandCaption','err');
btnAdm.Caption:=IniGetStringValue('StartUnit','btnAdmCaption','err');
BtnLoadNew.Caption:=IniGetStringValue('StartUnit','BtnLoadNewCaption','err');
btnReports.Caption:=IniGetStringValue('StartUnit','btnReportsCaption','err');
BitBtn8.Caption:=IniGetStringValue('StartUnit','BitBtn8Caption','err');

end;

```

```

procedure TfrmStart.addLog(Userlog, dast, dep, Where_, Wat_: string);
var
  dastup,depor: string;
begin
  if dast ='' then dastup :='null'
  else dastup:= dast;
  if dep ='' then depor :='null'
  else depor := dep;
  try
    with dm_budget.ADOQuery1 do begin
      close; //main
      sql.Text:=
        'insert into '+Log_Table+' (User_,      dastup,      dep,  Where_,      Wat_,
          Date_) '+
          'values ('+''+Userlog+''+', '+dastup+', '+depor+', '+'''+Where_+''+',
'+'''+Wat_+''+', '+getdate()'+)';
    end;
  except
  end;
end;

```

```

      execSql;
end;
Except
end;

```

```

end;

```

```

procedure TfrmStart.BitBtn8Click(Sender: TObject);
begin
  close;
end;

```

```

procedure TfrmStart.btn1EnterClick(Sender: TObject);
begin
  //-----
  frmStart.addLog(login, access_, depCode ,'StartUnit ', 'Открытие "Ввод и
корректировка данных (доход)");
  //-----
end;

```



```
depCode:='1';
```

```
frmDohod:=tfrmDohod.create(self);  
frmDohod.WindowState:=wsMaximized;  
frmDohod.Caption:=btn1Enter.Caption;  
frmDohod.ShowModal;  
end;
```

```
procedure TfrmStart.btnReportClick(Sender: TObject);
```

```
begin
```

```
//-----
```

```
frmStart.addLog(login, access_, depCode, 'StartUnit ', 'Открытие "Просмотр  
отчетов");
```

```
//-----
```

```
{1 Бюджет г.Алматы
```

```
2 Проект бюджета г.Алматы на ____ год
```

```
3 Утвержденный бюджет г.Алматы
```

```
4 Свод данных по уточнениям бюджета г.Алматы в ____ году
```

```
5 Сравнительная таблица по бюджету г.Алматы на ____ год (агрегированная)
```

```
6 Сравнительная таблица по бюджету г.Алматы за ____ - ____ годы (по  
кассовому исполнению)
```

```
7 Сравнительная таблица по бюджету г.Алматы за ____ - ____ годы (по  
удельному весу)
```

```
8 Сравнительная таблица по бюджету г.Алматы за ____ - ____ годы (по  
отраслям)
```

```
9 Уточненный бюджет г.Алматы на текущий год (по мероприятиям и видам  
работ)
```

```
10 Отчет об исполнении бюджета города республиканского значения,  
столицы (форма Департамента финансов)
```

```
11 Поступление налогов и других платежей в бюджет г.Алматы за (месяц*)
```

```
12 Собираемость доходов за (кол-во месяцев отчетного года на момент  
отчета) месяцев
```

```
13 Ежедневные поступления доходов за 3 месяца (текущего финансового  
года)
```

```
14 Сравнительная таблица плановых заданий и фактического исполнения  
прогнозов за ____ - ____ годы
```

```
15 Таблица долговых обязательств местного исполнительного органа  
г.Алматы (по состоянию на дд.мм.гггг года)
```

```
16 Справка-расчет по проектам займов
```

```
17 Сравнительная таблица по бюджету г.Алматы за ____ - ____ годы}
```

```

frmReport:=tfrmReport.create(self);
frmReport.WindowState:= wsMaximized;
frmReport.ShowModal;
end;

procedure TfrmStart.FormCreate(Sender: TObject);

begin
MyIniFile := TIniFile.Create('C:\Program Files\budget\Settings.ini');

frIniFile := TIniFile.Create('C:\Program Files\budget\LoadNewProgram\my.ini');
from_:=frIniFile.ReadString('data','from','\\192.168.7.1\bp\budget');
frIniFile.free;
frIniFile := TIniFile.Create(from_+'\Settings.ini');

    ///---определение директории--
    GetDir(0,sPath);
    ///---загрузка языка----
    langIni := TIniFile.Create(sPath +'\LANG.ini');
    ///---процедура перевода---
    LoadLang;

//-----autentification-----
frmAuthenticate:=tfrmAuthenticate.Create(self);
if frmAuthenticate.ShowModal = mrOK then begin //1

    end else begin //1
        Application.Terminate;
    end;

DecimalSeparator:= '.';
ShortDateFormat:= 'dd.mm.yyyy';
//инициализируем глобальные переменные
frmScreen:=tfrmScreen.Create(self);
frmScreen.ShowModal;
lang:=MyIniFile.ReadString('language', 'lang', 'rus');

image1.Picture.LoadFromFile('C:\Program Files\budget\picture\backGround.jpg');

end;

```

```

procedure TfrmStart.btnAdmClick(Sender: TObject);
begin
//-----
frmStart.addLog(login, access_, depCode , 'StartUnit ', 'Открытие
"Администрирование");
//-----

frmAdditional:=tfrmAdditional.Create(self);
frmAdditional.Caption:=btnAdm.Caption;
frmAdditional.ShowModal;
end;

procedure TfrmStart.FormClose(Sender: TObject; var Action: TCloseAction);
begin
//-----
frmStart.addLog(login, access_, depCode , 'StartUnit ', 'Выход ' + login);
//-----
MyIniFile.Free;
end;

procedure TfrmStart.btn2EnterClick(Sender: TObject);
var i:byte;
begin
//-----
frmStart.addLog(login, access_, depCode , 'StartUnit ', 'Открытие "Ввод и
корректировка данных (образование)");
//-----
depCode:='2';

gExpensesTable:='expenses_eduCultur';
gActionTable:='action_eduCultur';

gProcedureForShow:='showEduCulturBySp';
gProcedureForDelShow:='showDelEduCulturBySp';
gProcedureRestore:='restoreEduCultur';
gProcedureUpd:='updDevelopEduCultur';

```

```

countF_ex:=strToInt(MyIniFile.ReadString('expenses_eduCultur','countOfFieldsForVisible','0'));
point0_ex:=strToInt(MyIniFile.ReadString('expenses_eduCultur','pointAsZerro','0'));
setLength(name_of_column_ex,countF_ex);
setLength(caption_of_column_ex,countF_ex);

for i:=0 to countF_ex-1 do begin

name_of_column_ex[i]:=MyIniFile.ReadString('expenses_eduCultur','name_of_column'+intToStr(i),'***');

caption_of_column_ex[i]:=MyIniFile.ReadString('expenses_eduCultur','caption_of_column'+intToStr(i),'***');
end;

frmDepRealSector:=TfrmDepRealSector.Create(self);
frmDepRealSector.WindowState:=wsMaximized;
frmDepRealSector.Caption:=btn2Enter.Caption;
frmDepRealSector.showModal;
{ frmDepEduCultur:=TfrmDepEduCultur.Create(self);
frmDepEduCultur.WindowState:=wsMaximized;
frmDepEduCultur.showModal; }
end;

procedure TfrmStart.btn3EnterClick(Sender: TObject);
var i:byte;
begin

//-----
frmStart.addLog(login, access_, depCode , 'StartUnit ', 'Открытие "Ввод и
корректировка данных (соцзащита)");
//-----

gExpensesTable:='expenses_Social';
gActionTable:='action_Social';

```

```

gProcedureForShow:='showSocialBySp';
gProcedureForDelShow:='showDelSocialBySp';
gProcedureRestore:='restoreSocial';
gProcedureUpd:='updDevelopSocial';

countF_ex:=strToInt(MyIniFile.ReadString('expenses_Social','countOfFieldsForVisible','0'));
point0_ex:=strToInt(MyIniFile.ReadString('expenses_Social','pointAsZerro','0'));
setLength(name_of_column_ex,countF_ex);
setLength(caption_of_column_ex,countF_ex);

for i:=0 to countF_ex-1 do begin

name_of_column_ex[i]:=MyIniFile.ReadString('expenses_Social','name_of_column'+intToStr(i),'***');

caption_of_column_ex[i]:=MyIniFile.ReadString('expenses_Social','caption_of_column'+intToStr(i),'***');
end;

depCode:='3';
{ frmDepSocial:=TfrmDepSocial.Create(self);
frmDepSocial.WindowState:=wsMaximized;
frmDepSocial.showModal; }
//if not(Assigned(frmDepRealSector)) then begin
frmDepRealSector:=TfrmDepRealSector.Create(self);
frmDepRealSector.WindowState:=wsMaximized;
//end;
frmDepRealSector.Caption:=btn3Enter.Caption;
frmDepRealSector.showModal;
end;

procedure TfrmStart.btn5EnterClick(Sender: TObject);
var i:byte;
begin
//-----
frmStart.addLog(login, access_, depCode , 'StartUnit ', 'Открытие "Ввод и
корректировка данных (госаппарат)");
//-----

gActionTable:='action_GosApparat';

```

```

countF_ex:=strToInt(MyIniFile.ReadString('expenses_GosApparat','countOfFields
ForVisible','0'));
point0_ex:=strToInt(MyIniFile.ReadString('expenses_GosApparat','pointAsZerro',
0));
setLength(name_of_column_ex,countF_ex);
setLength(caption_of_column_ex,countF_ex);

for i:=0 to countF_ex-1 do begin

name_of_column_ex[i]:=MyIniFile.ReadString('expenses_GosApparat','name_of_
column'+intToStr(i,'***'));

caption_of_column_ex[i]:=MyIniFile.ReadString('expenses_GosApparat','caption_
of_column'+intToStr(i,'***'));
end;

depCode:='5';

{ frmDepGosApparat:=TfrmDepGosApparat.Create(self);
frmDepGosApparat.WindowState:=wsMaximized;
frmDepGosApparat.showModal; }

//*****
gExpensesTable:='expenses_GosApparat';
gActionTable:='action_GosApparat';

gProcedureForShow:='showGosApparatBySp';
gProcedureForDelShow:='showDelGosApparatBySp';
gProcedureRestore:='restoreGosApparat';
gProcedureUpd:='updDevelopGosApparat';

//if not(Assigned(frmDepRealSector)) then begin
frmDepRealSector:=TfrmDepRealSector.Create(self);
frmDepRealSector.WindowState:=wsMaximized;
//end;
frmDepRealSector.Caption:=btn5Enter.Caption;
frmDepRealSector.showModal;
//*****
end;

```

```

procedure TfrmStart.btn4EnterClick(Sender: TObject);
var i:byte;
begin
//-----
frmStart.addLog(login, access_, depCode , 'StartUnit ', 'Открытие "Ввод и
корректировка данных (транспорт)");
//-----
gExpensesTable:='expenses_Trans';
gActionTable:='action_Trans';

gProcedureForShow:='showTransBySp';
gProcedureForDelShow:='showDelTransBySp';
gProcedureRestore:='restoreTrans';
gProcedureUpd:='updDevelopTrans';

depCode:='4';
countF_ex:=strToInt(MyIniFile.ReadString(gExpensesTable,'countOfFieldsForVis
ible','0'));
point0_ex:=strToInt(MyIniFile.ReadString(gExpensesTable,'pointAsZerro','0'));
setLength(name_of_column_ex,countF_ex);
setLength(caption_of_column_ex,countF_ex);

for i:=0 to countF_ex-1 do begin

name_of_column_ex[i]:=MyIniFile.ReadString(gExpensesTable,'name_of_column
'+intToStr(i,'***'));

caption_of_column_ex[i]:=MyIniFile.ReadString(gExpensesTable,'caption_of_col
umn'+intToStr(i,'***'));
end;

frmDepRealSector:=TfrmDepRealSector.Create(self);
frmDepRealSector.WindowState:=wsMaximized;

frmDepRealSector.Caption:=btn4Enter.Caption;
frmDepRealSector.showModal;
end;

```

```

procedure TfrmStart.FormShow(Sender: TObject);
begin
//frmStart.WindowState:=wsMaximized;
btnReport.Visible:=true;
if access_='1' then begin
  btnAdm.Visible:=true;
  btn1Enter.Visible:=true;
  btn2Enter.Visible:=true;
  btn3Enter.Visible:=true;
  btn4Enter.Visible:=true;
  btn8Enter.Visible:=true;
  btn5Enter.Visible:=true;
  btn6Enter.Visible:=true;
frmStart.Caption:=frmStart.IniGetStringValue('StartUnit','FrmCaption','err');
btn1Enter.Caption:=IniGetStringValue('StartUnit','btn1Caption','err');
btn2Enter.Caption:=IniGetStringValue('StartUnit','btn2Caption','err');
btn3Enter.Caption:=IniGetStringValue('StartUnit','btn3Caption','err');
btn4Enter.Caption:=IniGetStringValue('StartUnit','btn4Caption','err');
btn5Enter.Caption:=IniGetStringValue('StartUnit','btn5Caption','err');
btn6Enter.Caption:=IniGetStringValue('StartUnit','btn6Caption','err');
btn8Enter.Caption:=IniGetStringValue('StartUnit','btn8Caption','err');

  btn1Enter.Width:=550;
  btnReport.Width:=550;
  btnAdm.Width:=550;
  btn2Enter.Width:=550;
  btn3Enter.Width:=550;
  btn4Enter.Width:=550;
  btn5Enter.Width:=550;
  btn6Enter.Width:=550;
  btn8Enter.Width:=550;
  BtnLoadNew.Width:=550;
  btnDemand.Width:=550;
  exit; // 609
end;

btnReport.Visible:=true;
btnReport.Top:=48;
BtnLoadNew.Top:=96;
btnDemand.Top:=136;
{if access_='4' then begin
  frmReport:=tfrmReport.Create(self);
  frmReport.WindowState:=wsMaximized;

```



```
frmReport.ShowModal;  
close;  
end; }
```

```
if deocode='1' then begin  
  btn1Enter.Visible:=true;  
  btn1Enter.Caption:=IniGetStringValue('StartUnit','btnCaption','err');  
  btn1Enter.Top:=8;  
end;
```

```
if deocode='2' then begin  
  btn2Enter.Visible:=true;  
  btn2Enter.Caption:=IniGetStringValue('StartUnit','btnCaption','err');  
  btn2Enter.Top:=8;  
end;
```

```
if deocode='3' then begin  
  btn3Enter.Visible:=true;  
  btn3Enter.Caption:=IniGetStringValue('StartUnit','btnCaption','err');  
  btn3Enter.Top:=8;  
end;
```

```
if deocode='4' then begin  
  btn4Enter.Visible:=true;  
  btn4Enter.Caption:=IniGetStringValue('StartUnit','btnCaption','err');  
  btn4Enter.Top:=8;  
end;
```

```
if deocode='5' then begin  
  btn5Enter.Visible:=true;  
  btn5Enter.Caption:=IniGetStringValue('StartUnit','btnCaption','err');  
  btn5Enter.Top:=8;  
end;
```

```
if deocode='6' then begin  
  btn6Enter.Visible:=true;  
  btn6Enter.Caption:=IniGetStringValue('StartUnit','btnCaption','err');  
  btn6Enter.Top:=8;  
end;
```

```
if deocode='9' then begin  
  btn9Enter.Visible:=true;  
  btn9Enter.Caption:=IniGetStringValue('StartUnit','btnCaption','err');  
  btn9Enter.Top:=8;
```

```
end;
if deocode='8' then begin
  btn8Enter.Visible:=true;
  btn8Enter.Caption:=IniGetStringValue('StartUnit','btnCaption','err');
  btn8Enter.Top:=8;
end;
```

```
if deocode='7' then begin
  frmConsolReports:=tfrmConsolReports.Create(self);
  frmConsolReports.WindowState:=wsMaximized;
  frmConsolReports.ShowModal;
  close;
end;
end;
```

```
procedure TfrmStart.BtnLoadNewClick(Sender: TObject);
begin
  //-----
  frmStart.addLog(login, access_, depCode, 'StartUnit ', 'Загрузка обновления
  программы');
  //-----
  if not DirectoryExists('c:\temp') then
    createDir('c:\temp') ;
  copyFile('C:\Program
  Files\budget\LoadNewProgram\LoadNew.exe','c:\temp\LoadNew.exe',false);
  winExec('c:\temp\LoadNew.exe',1);
  close;
end;
```

```
procedure TfrmStart.readXmlClick(Sender: TObject);
begin
  frmReadXml:=tfrmReadXml.create(self);
  frmReadXml.ShowModal;
end;
```

```

procedure TfrmStart.btnReportsClick(Sender: TObject);
begin
//-----
frmStart.addLog(login, access_, depCode , 'StartUnit ', 'Открытие "5 отчетов"');
//-----

frmConsolReports:=tfrmConsolReports.Create(self);
frmConsolReports.WindowState:=wsMaximized;
frmConsolReports.ShowModal;
end;

```

```

procedure TfrmStart.btn6EnterClick(Sender: TObject);
var i:byte;
begin

//-----
frmStart.addLog(login, access_, depCode , 'StartUnit ', 'Открытие "Ввод и
корректировка данных (сводный)");
//-----
{countF_ex:=strToInt(MyIniFile.ReadString('expenses_Svod','countOfFieldsForVi
sible','0'));
point0_ex:=strToInt(MyIniFile.ReadString('expenses_Svod','pointAsZerro','0'));
setLength(name_of_column_ex,countF_ex);
setLength(caption_of_column_ex,countF_ex);

for i:=0 to countF_ex-1 do begin

name_of_column_ex[i]:=MyIniFile.ReadString('expenses_Svod','name_of_column
'+intToStr(i),'***');

caption_of_column_ex[i]:=MyIniFile.ReadString('expenses_Svod','caption_of_col
umn'+intToStr(i),'***');
end;

depCode:='6';
frmDepSvod:=TfrmDepSvod.Create(self);
frmDepSvod.WindowState:=wsMaximized;

```

```
frmDepSvod.Caption:=btn6Enter.Caption;  
frmDepSvod.showModal; }
```

```
gExpensesTable:='expenses_Svod';
```

```
gProcedureForShow:='showSvodBySp';  
gProcedureForDelShow:='showDelSvodBySp';  
gProcedureRestore:='restoreSvod';  
gProcedureUpd:='updDevelopSvod';
```

```
depCode:='6';  
countF_ex:=strToInt(MyIniFile.ReadString(gExpensesTable,'countOfFieldsForVisible','0'));  
point0_ex:=strToInt(MyIniFile.ReadString(gExpensesTable,'pointAsZerro','0'));  
setLength(name_of_column_ex,countF_ex);  
setLength(caption_of_column_ex,countF_ex);
```

```
for i:=0 to countF_ex-1 do begin
```

```
name_of_column_ex[i]:=MyIniFile.ReadString(gExpensesTable,'name_of_column'+intToStr(i),'***');
```

```
caption_of_column_ex[i]:=MyIniFile.ReadString(gExpensesTable,'caption_of_column'+intToStr(i),'***');  
end;
```

```
frmDepRealSector:=TfrmDepRealSector.Create(self);  
frmDepRealSector.WindowState:=wsMaximized;
```

```
frmDepRealSector.Caption:=btn6Enter.Caption;  
frmDepRealSector.showModal;  
end;
```

```
procedure TfrmStart.btnDemandClick(Sender: TObject);  
begin
```

```
//-----
```

```
frmStart.addLog(login, access_, depCode, 'StartUnit ', 'Открытие "Учет бюджетных заявок");
```

```
//-----
```

```
frmSelectZ:=TfrmSelectZ.Create(self);
```

```
frmSelectZ.Caption:=btnDemand.Caption;
frmSelectZ.ShowModal;
end;
```

```
procedure TfrmStart.Button1Click(Sender: TObject);
var f1, f2:string;
begin
with dm_budget.adoQuery1 do begin
close;
sql.Text:='select RussianName, KazakhName '+
' FROM sp_outcomes_abp where actual="1"';
open;
while not eof do begin
if pos('Департамент', fields[0].asString)>0 then begin
edit;
f1:=fields[0].asString;
f2:=fields[1].asString;
system.delete(f1, pos('Департамент',f1), length('Департамент'));
f1:='Управление'+f1;
system.delete(f2, pos('департаменті',f2), length('департаменті'));
f2:=f2+'басқармасы';
fields[0].asString:=f1;
fields[1].asString:=f2;
post;
end;
next;
end;
end;
end;
```

```
{with dm_budget.adoQuery1 do begin
close;
sql.Text:='update sp_outcomes_subprogram'+
' set KazakhName="'+trim(edit2.Text)+'" '+
// set RussianName="'+trim(edit2.Text)+'" '+
' where subprogram="'+trim(edit1.Text)+'"';
execSql;
close;
sql.Text:='select KazakhName from sp_outcomes_subprogram'+
' where subprogram="'+trim(edit1.Text)+'"';
open;
edit3.Text:=fields[0].AsString;
end; }
end;
```

```

procedure TfrmStart.btn9EnterClick(Sender: TObject);
var i:byte;
begin
//-----
frmStart.addLog(login, access_, depCode, 'StartUnit ', 'Открытие "Ввод и
корректировка данных (всего)");
//-----
gExpensesTable:='expenses_all';
gActionTable:='action_all';

gProcedureForShow:='showallBySp';
gProcedureForDelShow:='showDelallBySp';
gProcedureRestore:='restoreall';
gProcedureUpd:='updDevelopall';

depCode:='17';
countF_ex:=strToInt(MyIniFile.ReadString(gExpensesTable,'countOfFieldsForVis
ible','0'));
point0_ex:=strToInt(MyIniFile.ReadString(gExpensesTable,'pointAsZerro','0'));
setLength(name_of_column_ex,countF_ex);
setLength(caption_of_column_ex,countF_ex);

for i:=0 to countF_ex-1 do begin

name_of_column_ex[i]:=MyIniFile.ReadString(gExpensesTable,'name_of_column
'+intToStr(i,'***'));

caption_of_column_ex[i]:=MyIniFile.ReadString(gExpensesTable,'caption_of_col
umn'+intToStr(i,'***'));
end;

frmDepRealSector1:=TfrmDepRealSector1.Create(self);
frmDepRealSector1.WindowState:=wsMaximized;

frmDepRealSector1.Caption:=btn8Enter.Caption;
frmDepRealSector1.showModal;
end;

procedure TfrmStart.btn8EnterClick(Sender: TObject);
var i:byte;
begin

```

```

//-----
frmStart.addLog(login, access_, depCode, 'StartUnit ', 'Открытие "Ввод и
корректировка данных (благоустройства)");
//-----
gExpensesTable:='expenses_Stroit';
gActionTable:='action_Stroit';

gProcedureForShow:='showStroitBySp';
gProcedureForDelShow:='showDelStroitBySp';
gProcedureRestore:='restoreStroit';
gProcedureUpd:='updDevelopStroit';

depCode:='8';
countF_ex:=strToInt(MyIniFile.ReadString(gExpensesTable,'countOfFieldsForVis
ible','0'));
point0_ex:=strToInt(MyIniFile.ReadString(gExpensesTable,'pointAsZerro','0'));
setLength(name_of_column_ex,countF_ex);
setLength(caption_of_column_ex,countF_ex);

for i:=0 to countF_ex-1 do begin

name_of_column_ex[i]:=MyIniFile.ReadString(gExpensesTable,'name_of_column
'+intToStr(i),'***');

caption_of_column_ex[i]:=MyIniFile.ReadString(gExpensesTable,'caption_of_col
umn'+intToStr(i),'***');
end;

frmDepRealSector:=TfrmDepRealSector.Create(self);
frmDepRealSector.WindowState:=wsMaximized;

frmDepRealSector.Caption:=btn8Enter.Caption;
frmDepRealSector.showModal;
end;

procedure TfrmStart.BitBtn8Exit(Sender: TObject);
begin
//-----
frmStart.addLog(login, access_, depCode, 'StartUnit ', 'ВЫХОД');
//-----
end;

procedure TfrmStart.BitBtnlangClick(Sender: TObject);

```

```
begin
  LangForm:=tLangForm.create(self);

  LangForm.Caption:=IniGetStringValue('langUnit','FrmCaption','err');
  LangForm.ShowModal;
end;

end.
```