

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ
ИМЕНИ ГУМАРБЕКА ДАУКЕЕВА»

Кафедра «IT – инжиниринг»

«ДОПУЩЕН К ЗАЩИТЕ»

Зав. кафедрой _____

(ученая степень, звание, Ф.И.О.)

_____ «_____» _____ 202__ г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Разработка информационного портала для компании. Разработка системы хранения и обработки цифровых данных

Специальность 5B070400 - Вычислительная техника и программное обеспечение

Выполнил(а) Бобровский П. А. _____ Группа BT-16-2
(Ф.И.О.)

Научный руководитель PhD, доцент Кожамкулова Ж.Ж.
(учёная степень, звание,
Ф.И.О.)

Консультанты:
по экономической части: проф., к.э.н. Габелашвили К.Р.
(учёная степень, звание,
Ф.И.О.)

_____ «_____» _____ 202__ г.
по безопасности жизнедеятельности: доцент, к.т.н. Приходько Н.Г.
(учёная степень, звание,
Ф.И.О.)

_____ «_____» _____ 202__ г.
по программному обеспечению: ст. преп. Майкотов М.Н.
(учёная степень, звание,
Ф.И.О.)

_____ «_____» _____ 202__ г.
Нормоконтролер: ст. преп. Абсатарова Б.Р.

Ф.И.О.) _____ (учёная степень, звание,
« ____ » _____ 202__ г.

Рецензент: _____

Ф.И.О.) _____ (учёная степень, звание,
« ____ » _____ 202__ г.

Алматы 2020

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН
Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ
ИМЕНИ ГУМАРБЕКА ДАУКЕЕВА»

Институт систем управления и информационных технологий

Специальность 5В070400 - Вычислительная техника и программное обеспечение

Кафедра IT-инжиниринг

ЗАДАНИЕ

на выполнение дипломного проекта

Студенту Бобровскому Павлу Александровичу

(фамилия, имя, отчество)

Тема работы (проекта) Разработка информационного портала для компании.
Разработка системы хранения и обработки цифровых данных

Утверждена приказом по университету № ___ от «___» _____ 202_ г.

Срок сдачи законченного проекта «___» _____ 202_ г.

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

- 1) Провести анализ предметной области;
- 2) Разработать техническое задание (ТЗ);
- 3) Выполнить сравнительный анализ существующих аналогов;
- 4) Выбрать программное обеспечение;
- 5) Создать логическую и физическую модели данных;
- 6) Создать UML диаграммы.

Перечень вопросов, подлежащих разработке в дипломном проекте, или краткое содержание дипломного проекта:

- 1) анализ предметной области информации, подлежащей хранению и обработке в базе данных;
- 2) проектирование базы данных, ориентированной на предметную область;
- 3) процесс создания базы данных.

Перечень графического материала (с точным указанием обязательных чертежей):

- 1) Физическая модель базы данных;
- 2) Логическая модель базы данных;
- 3) UML диаграммы;
- 4) Презентация (21 слайдов).

Основная рекомендуемая литература:

- 1) MySQL руководство администратора. - М.: Вильямс, 2017. - 621 с.
- 2) Яргер, Р.Дж. MySQL и mSQL: Базы данных для небольших предприятий и Интернета / СПб: Символ-Плюс, 2017. - 560 с.
- 3) Аткинсон, Леон MySQL. Библиотека профессионала; М.: Вильямс, 2018. - 624 с.

Консультация по проекту с указанием относящихся к ним разделов проекта

Раздел	Консультант	Сроки	Подпись
Экономическая часть	Габелашвили К.Р.	01.04 - 18.04.20	
БЖД	Приходько Н.Г.	04.04 - 23.04.20	
Нормоконтроль	Абсатарова Б.Р.	25.04 - 08.05.20	
ПО	Майкотов М.Н.	27.04 - 11.05.20	

ГРАФИК

подготовки дипломной работы (проекта)

Наименования разделов, перечень разрабатываемых вопросов	Сроки представления научному руководителю	Примечания
Анализ предметной области	13.01.20	
Цель разработки	21.01.20	
Сравнительный анализ ПО	04.02.20	
Выбор ПО для разработки	18.02.20	
Разработка технического задания	02.03.20	
Построение лог. и физ. моделей	30.03.20	
Построение UML диаграмм	07.04.20	
Создание базы данных	14.04.20	
Оформление отчета	12.05.20	
Создание презентации	16.05.20	

Дата выдачи задания «13» января 2020г.

Заведующий кафедрой _____ (_____)

(подпись)

(Ф.И.О.)

Научный руководитель
проекта

_____ (_____)
(подпись)

(Ф.И.О.)

Задание принял к
исполнению студент

_____ (_____)
(подпись)

(Ф.И.О.)

АНДАТПА

Бұл дипломдық жобаның мақсаты Гүлдерді көтерме және бөлшек сатуды жүзеге асыратын флористикалық компания үшін сандық деректерді сақтау және өңдеу жүйесінің модулін құру болып табылады.

Бұл дипломдық жоба тақырыбының өзектілігі қызметкерлердің қызметін басқаруды жүзеге асыратын флористикалық компанияның қызметі жағдайында әртүрлі сандық деректерді сақтау және өңдеу қажеттілігімен байланысты.

Сонымен қатар, соңғы екі тарауларда тіршілік қауіпсіздігі мәселелері қарастырылады, техникалық-экономикалық негіздеме келтіріледі, сондай-ақ әзірлеу бағасы, іске асыру құны және жобаның өтелу мерзімі есептеледі.

АННОТАЦИЯ

Целью данного дипломного проекта является создание модуля системы хранения и обработки цифровых данных для флористической компании, осуществляющей оптовые и розничные продажи цветов.

Актуальность темы данного дипломного проекта связана с необходимостью хранения и обработки разного рода цифровых данных в условиях деятельности флористической компании, осуществляющего управление деятельностью сотрудников.

Помимо этого, в двух последних главах рассматриваются вопросы безопасности жизнедеятельности, приводится технико-экономическое обоснование, а также рассчитывается цена разработки, стоимость реализации и срок окупаемости проекта.

ANNOTATION

The purpose of this diploma project is to create a module for storing and processing digital data for a florist company that sells wholesale and retail flowers.

The relevance of the topic of this diploma project is related to the need to store and process various types of digital data in the conditions of a florist company that manages the activities of employees.

In addition, the last two chapters address issues of life safety, provide a feasibility study, and calculate the cost of development, cost of implementation, and payback period of the project.

Содержание

Введение.....	8
1 Аналитическая часть.....	9
1.1 Описание предметной области.....	9
1.2 Цель разработки.....	10
1.3 Обоснование выбора ПО.....	10
1.4 Сравнительный анализ СУБД.....	12
2 Проектная часть.....	15
2.1 Техническое задание.....	15
2.2 Логическая и физическая модели базы данных.....	15
2.3 UML диаграммы базы данных.....	18
3 Экспериментальная часть.....	21
3.1 Проектирование базы данных.....	21
3.2 Программная реализация базы данных.....	23
4 Безопасность жизнедеятельности.....	37
4.1 Анализ потенциально опасных и вредных факторов в офисе, воздействующих на персонал.....	37
4.2 Расчет естественного и искусственного освещения помещения.....	40
4.3 Расчет системы кондиционирования офисного помещения.....	45
4.4 Вывод по разделу безопасность жизнедеятельности.....	49
5 Техничко-экономическое обоснование проекта.....	50
5.1 Цель проекта.....	50
5.2 Реализация проекта.....	50
5.3 Расчёт стоимости произведенного проекта.....	53
5.4 Расчёт затрат на оплату труда.....	53
5.5 Цена реализации программного продукта.....	59
5.6 Расчет эффективности и срока окупаемости программного продукта.....	60
5.7 Вывод по технико-экономическому разделу.....	64
Заключение.....	65
Список литературы.....	66
Приложение А SQL код создания БД.....	67
Приложение Б Акт внедрения.....	72

ВВЕДЕНИЕ

Информация в современном мире является одним из наиболее важных ресурсов, а БД (базы данных) превратились в один из необходимых инструментов для хранения этой же самой информации почти во всех сферах деятельности.

Любая современная компания не может обойтись без базы данных. Базы данных позволяют им хранить всю необходимую информацию о действующих сотрудниках компании, принимать и отправлять различного рода документы и/или отчеты, хранить и отслеживать текущие статусы поставленных задач, а также использоваться в качестве хранилища текстовых сообщений внутреннего чата между сотрудниками компании.

Конечно, снижение времени является лишь побочным эффектом автоматизации. Самая главная задача развития информационных технологий в совсем другом - в приобретении той или иной компанией исключительно новых качеств, придающих ей существенную конкурентоспособность. А это дорогого стоит.

К тому же, в настоящее время установка и управление базами данных не является таким уж и сложным и трудным процессом, как это было десятилетие назад. В то время управление и проектировка баз данных были не автоматизированным и достаточно трудоемким процессом. Сейчас же, система управления базами данных позволяет создавать базу данных для любых целей, а также может обновлять в ней хранимую информацию, обеспечивать оперативный доступ к самой структуре БД для просмотра и поиска информации.

Целью данного дипломного проекта является разработка системы хранения и обработки цифровых данных для флористической компании. В ходе выполнения дипломной работы будут затронуты следующие важные вопросы:

- провести анализ предметной области;
- разработать техническое задание (ТЗ);
- выполнить сравнительный анализ существующих аналогов;
- выбрать программное обеспечение для реализации базы данных;
- спроектировать базу данных;
- создать логическую и физическую модели баз данных;
- создать UML диаграммы;
- осуществить программную реализацию базы данных.

Помимо всего вышеперечисленного, будут затронуты раздел безопасности жизнедеятельности, в котором будет рассчитано необходимое естественное и искусственное освещения, и нормирование показателей микроклимата для офисного помещения, а также технико-экономическое обоснование, на основе которого будут рассчитаны цена на разработку, стоимость реализации и срок окупаемости разработанного проекта.

1 Аналитическая часть

1.1 Описание предметной области

Развитие IT (информационных технологий) в данный момент времени происходит очень стремительно. Практически в любой сфере бизнеса, хозяйственной или же финансовой деятельности имеется своя система хранения и обработки цифровых данных. Например, для информационного портала флористической компании использование базы данных позволит упростить и автоматизировать работу всех внутренних процессов.

В нынешних условиях в средних и крупных компаниях сотрудники вынуждены иметь дело с большим количеством важной и часто изменяющейся информации, которую еще нужно где-то хранить. В таких компаниях, с большим количеством сотрудников и бумажным документооборотом, существует необходимость управления внушительным объемом кадровой информации (актуальные данные всех действующих сотрудников), различного рода документами (документы на увольнение, принятие на работу, отпуск, премиальные и т.д.), хранения текстовых сообщений сотрудников (внутренний мессенджер), а также хранение разного рода поставленных и выполненных задач, включая их текущие статусы.

Тема дипломной работы: «Разработка информационного портала для компании. Разработка системы хранения и обработки цифровых данных». Актуальность темы данного дипломного проекта связана с необходимостью хранения и обработки разного рода цифровых данных в условиях деятельности флористической компании, которая осуществляет управление деятельностью сотрудников и документооборотом.

Использование системы хранения и обработки в этой сфере позволяет решить комплекс задач, связанных с оптимизацией и автоматизацией обработки цифровых данных и документооборота, которые используются внутри компании, а также решить проблемы с их хранением и последующим использованием. Результатом выполнения вышеперечисленных задач является минимизация затрат компании, уменьшение объема работы сотрудников компании, улучшение их взаимодействия друг с другом, а также повышение производительности работоспособности сотрудников.

Пользователями данного программного продукта являются действующие сотрудники флористической компании.

1.2 Цель разработки

Целью дипломного проекта является разработка модуля системы хранения и обработки цифровых данных для флористической компании, который предоставляет следующий функционал:

- хранение актуальной информации о сотрудниках;
- хранение разного рода документов (на принятие на работу, увольнение, отпуск и премию);
- хранение и отслеживание текущих статусов поставленных задач и их самих;
- хранение сообщений из внутреннего чата в текстовом виде;
- хранение курса валют.

1.3 Обоснование выбора ПО

В ходе выполнения данной работы я использовал базу данных MySQL, т.к. она соответствовала всем необходимым требованиям и является одной из самых популярных БД, использующихся на серверах хостингов. В качестве утилиты для удаленного взаимодействия с базой данных, я использовал веб-приложение phpmyadmin, которое является самым оптимальным и выигрышным вариантом среди своих аналогов в бесплатном сегменте.

Начнем с обзора БД (базы данных) созданной в MySQL, MySQL поддерживает язык запросов SQL и может использоваться в качестве SQL-сервера. Данный язык многофункционален и с его помощью можно исправлять, создавать и удалять ненужные данные из таблиц, выполнять запрос на поиск нужной информации, а также создавать сами таблицы и т.д. Вся суть SQL заключается в следующем алгоритме: клиент отправляет серверу запрос, сервер его обрабатывает и возвращает клиенту только те данные, которые были выбраны в результате работы этого запроса. Тем самым клиенту не нужно скачивать сами данные и использовать лишние вычисления, как, например, в утилите Microsoft Access. Также стоит отметить, что MySQL - это ПО с открытым исходным кодом, т.е. его можно свободно редактировать. Данный пакет распространяется на условиях лицензии GPL, и его можно бесплатно загрузить из Интернета для бесплатного применения. Как только появился Интернет и связанные с ним технологии, благодаря которым создаются Web-страницы, стремительно возрос спрос и на СУБД, которые бы отличались своим быстродействием, надежностью и стабильностью. И в этом аспекте отлично проявила себя СУБД MySQL, которая получилась весьма быстрой, простой и надежной в плане использования [1].

Для того чтобы, как реляционные базы данных хранят эти самые данные, лучше всего объяснить на примере типичной таблицы, содержащей строки и

столбцы, куда внесены ФИО, а также соответствующие им телефонные номера, местожительство и прочая полезная информация. Похожий вид имеет и реляционная база данных. То есть, в каждом столбце содержится определенное его имя, и все значения, находящиеся в нем, являются однотипными переменными. Столбцы строго сортируются, в отличие от строк, которые могут в себе содержать значения из разных таблиц. Выполнив запрос к базе данных, пользователь получит необходимый результат. Абсолютно все значения в базе данных делятся на две группы: уникальные и неуникальные. К первой группе относятся ИИН, номера банковских счетов, номера банковских карт и т.д. А к неуникальным относится информация, которая может дублироваться, к примеру: дата рождения, имя и пр.

Главными достоинствами MySQL, благодаря которым данный пакет нравится многим веб-программистам, являются довольно высокая скорость обработки данных и многофункциональность СУБД. Также она совершенно бесплатна и доступна любому пользователю, что тоже является огромным плюсом. Множество современных казахстанских хостингов используют именно MySQL. От других баз данных MySQL отличается тем, что она без проблем взаимодействует с интерфейсом API. С помощью данного интерфейса, юзер легко получает доступ из необходимой программы к системе управления БД (базами данных), даже если она написана на таких языках, как C, C# и Perl и т.п.

Администраторами баз данных и веб-сайтов, чаще всего используется сочетание MySQL с PHP. Большое количество систем (CMS) написаны именно на базе этого «тандема». Одним из ярких примеров является WordPress, который известен любому начинающему программисту. В WordPress используется огромное множество полезных функций, на основе которых обеспечивается взаимодействие с сервером MySQL, к примеру, «mysqli_connect» [2].

Теперь перейдем к обзору веб-приложения phpmyadmin. Phpmyadmin - Это веб-панель, использующаяся для управления базами данных MySQL. Данная утилита написана на языке программирования PHP. На данный момент является самой популярной утилитой для управления базами данных в MySQL. phpmyadmin применяется практически во всех хостинговых провайдерах. В действительности ничего трудного в использовании данной панели нет, т.к. интерфейс интуитивно понятный и к тому же существует русская локализация, которую можно сразу при входе переключить на любой доступный язык. Однако, для обыкновенного юзера, phpMyAdmin практически бесполезна. Но всё же она имеет ряд достоинств, которые играют немаловажную роль в ее большой популярности. Во-первых, редактирование данных очень удобно, в то время как, другие аналоги не дают возможность изменять какие-либо данные в БД. К тому же, можно выполнить резервное копирование базы данных достаточно быстро, причем можно выполнить экспорт как полной базы, так и частями, что просто замечательно, когда БД очень объемная. Также частями в неё можно

импортировать данные, например, после краха или DDoS атаки, что гораздо облегчает работу при больших объемах важной информации.

Для разработчиков веб-сайтов и веб-приложений, phpMyAdmin дает весьма внушительные возможности в управлении базами данных, такие, как создание самой базы данных, создание в ней нужных таблиц и полей, а также их последующая корректировка, если что-то не устроило, также можно наполнить данными или исправлениями уже существующие таблицы, которые хранятся в БД (базах данных). В ней присутствует простейшая панель для вбивания SQL команд, панель для запроса по выбранному шаблону, и также имеется встроенный поиск по выбранной БД. Имеется встроенный менеджер структуры базы данных, который позволяет осуществлять визуальный мониторинг и редактирование таблиц MySQL. Имеются и инструменты для импорта и экспорта всех данных, либо же какой-то их части, утилита для мониторинга текущего состояния MySQL (например, использование ОЗУ и нагрузка процессора), также содержатся данные о серверных переменных и настройках сервера. Иными словами, в панели управления phpMyAdmin есть все, что нужно для рядового программиста, который, благодаря вышеназванным достоинствам, сможет легко и быстро управлять БД [3].

Однако, множество функций в phpMyAdmin не уникальны. К примеру, выполнять запросы можно и в утилите MySQL Query Browser, в которой уже присутствует полноценный отладчик SQL. Выполнять администрирование сервера, тонко настраивать его и управлять текущим мониторингом производительности можно с помощью программы MySQL Administrator – такого обширного количества опций и настроек сервера баз данных не может предоставить ни одна из ныне существующих утилит. Импортирование и экспортирование данных, а также создание резервных копий баз данных, лучше всего производить в утилитах компании EMS. Но, у всех этих утилит есть один очень существенный недостаток – абсолютно все они предназначены для локальной установки (хотя и могут управлять сервером удаленно), большинство из них функционируют только под управлением ОС Windows (хотя и есть отдельные версии и под Linux, либо же Ubuntu). В заключении стоит отметить, что утилита phpMyAdmin позволяет сделать практически все, в чем нуждается обычный администратор баз данных, разработчик ПО или же рядовой пользователь – и выполнить это без необходимости установки какого-либо лишнего ПО, благодаря этому утилита phpmyadmin является одним из самых популярных бесплатных веб-приложений.

1.4 Сравнительный анализ СУБД

Ввиду того, что наполнением базы данных будет заниматься человек, который, возможно, не имеет даже самых простых навыков программирования, то

наиболее оптимальным и выгодным будет использование базы данных с простым и интуитивно понятным интерфейсом, такой как MySQL. Наиболее подходящим, простым и надёжным способом хранения данных пользователя являются базы данных (БД). Помимо универсальности и распространённости, СУБД MySQL обладает целым рядом существенных преимуществ перед другими базами данных. В частности, следующие особенности:

1) *Простота в работе.* MySQL очень легко устанавливается, а наличие плагинов и вспомогательных приложений (например, phpmyadmin) упрощает работу с созданием таблиц и их заполнением, а также самой базы [4].

2) *Безопасность и использование.* MySQL применяется для работы в сети и предоставляется практически всеми хостингами. Таким образом, к серверу MySQL можно подключаться из любой точки мира через веб-приложение (такое как, phpmyadmin), установленную на компьютере или ноутбуке юзера. СУБД MySQL также поддерживает шифрование соединений с использованием криптографического протокола SSL для обеспечения дополнительной защиты.

3) *Масштабируемость.* Выбранная СУБД является очень универсальной, поэтому MySQL в равной степени легко используется для работы с малыми и большими объемами данных.

4) *Скорость.* За счет упрощения используемых в ней стандартов SQL, достигается весьма высокая производительность.

5) *Поддержка языка запросов.* MySQL может выполнять все команды языка запросов SQL, который применяется во всех нынешних СУБД.

6) *Аппаратная совместимость.* СУБД MySQL может работать под управлением самых различных ОС (операционных систем), а именно: UNIX, Windows и OS/2. СУБД MySQL также может работать, как и на стационарных домашних ПК (персональных компьютерах), так и на мощных серверах хостинговых компаний.

7) *Небольшой размер.* СУБД MySQL имеет очень малый размер, особенно в сравнении с достаточно объемным дисковым пространством, который необходим большинству коммерческих СУБД (Например, чтобы развернуть данную БД потребуется: сервер Apache (23 Мбайт), СУБД MySQL (120 Мбайт) и phpmyadmin (150 Мбайт), что в сумме меньше 500 Мбайт).

8) *Неограниченные возможности доступа.* Данная СУБД позволяет одновременно подключаться неограниченному количеству юзеров, которые будут одновременно работать с базой данных [5].

Далее, на основе всех вышеприведенных достоинств СУБД MySQL, рассмотрим и проведем сравнительный анализ самых популярных систем управления базами данных, а именно следующие СУБД: MySQL, PostgreSQL, SQLite. Сравнительный анализ будет выполняться по следующим критериям: поддержка веб-хостингами, простота разработки и администрирования и надёжность хранения данных (таблица 1.1):

Таблица 1.1 – Сравнительный анализ СУБД

	MySQL	PostgreSQL	SQLite
Архитектура	Клиент-сервер	Клиент-сервер	Встраиваемая БД
Поддержка веб-хостингами (веб-серверами)	Очень хорошая. Подавляющее большинство хостингов PHP имеют поддержку MySQL	Удовлетворительная. Поддержка хостингами PostgreSQL встречается часто	Удовлетворительная. Поддержка SQLite хостингами часто встречается, можно направить запрос в поддержку об установке необходимого модуля
Простота разработки и администрирования	Администрирование через консоль, множество приложений для разных платформ для локального администрирования, администрирование через веб-приложение phpmyadmin.	Администрирование через консоль, множество приложений для разных платформ для локального администрирования, администрирование через веб-сервер pgadmin.	Администрирование через консоль, множество приложений для разных платформ для локального администрирования, администрирование через веб-сервер.
Надёжность хранения данных	Надёжность обеспечивается сервером баз данных. Имеются механизмы резервного копирования	Надёжность обеспечивается сервером баз данных. Имеются механизмы резервного копирования.	Вся база хранится в одном файле.

Ввиду того, что наш информационный портал требует особых условий к скорости работы БД, масштабируемости и др., то мы можем использовать любой из вариантов. Отдадим предпочтение MySQL ввиду её высокой популярности, достаточно высокой скорости работы, а также наличию большого количества документации, постоянными обновлениями и большого комьюнити разработчиков.

2 Проектная часть

2.1 Техническое задание

Суть технического задания от компании заключалась в следующем:

- создать на арендуемом хостинге базу данных, удовлетворяющую особым условиям компании к скорости работы БД, масштабируемости и др.;
- создать таблицы для хранения данных о сотрудниках и их должностях;
- создать таблицы для хранения данных для каждого из трех типов регистрации на информационном портале;
- создать таблицы для данных, получаемых при оформлении документов на принятие на работу, увольнение, премиальные и отпуск;
- создать таблицы для текстовых сообщений по внутреннему чату между сотрудниками;
- создать таблицы для хранения всех поставленных и выполненных задач и их статусы;
- создать таблицы для хранения текущего курса иностранных валют по отношению к тенге;
- создать таблицы для хранения данных, получаемых с сервера 1С.

2.2 Логическая и физическая модели базы данных

Логическая и физическая модели данных были созданы в программе AllFusion ERwin Data Modeler 7 (версия 7.0).

ERwin — графический инструмент для создания моделей баз данных и поддержки этих самых баз, хранилищ данных и витрин, а также моделей использования ресурсов данных компаний. Модели, созданные в ERwin, используются для визуализации структуры данных, а также чтобы облегчить организацию и управление данными, упростить сложные взаимосвязи данных, в особенности технологии создания баз данных и их сред развертывания.

ERwin имеет два уровня представления модели баз данных: физический и логический. На логическом уровне представления данные с конкретной СУБД никак не связаны, именно поэтому этот уровень представления может быть наглядно представлен даже для обычного сотрудника, который никак не связан с управлением и администрированием баз данных. Физический уровень данных — представляет собой отображение системного каталога, который в свою очередь зависит от конкретной реализации СУБД и для рядового сотрудника данный уровень представления может быть непонятен [6]. Ниже представлены схемы физической и логической модели данных (рисунки 2.1 и 2.2 соответственно):

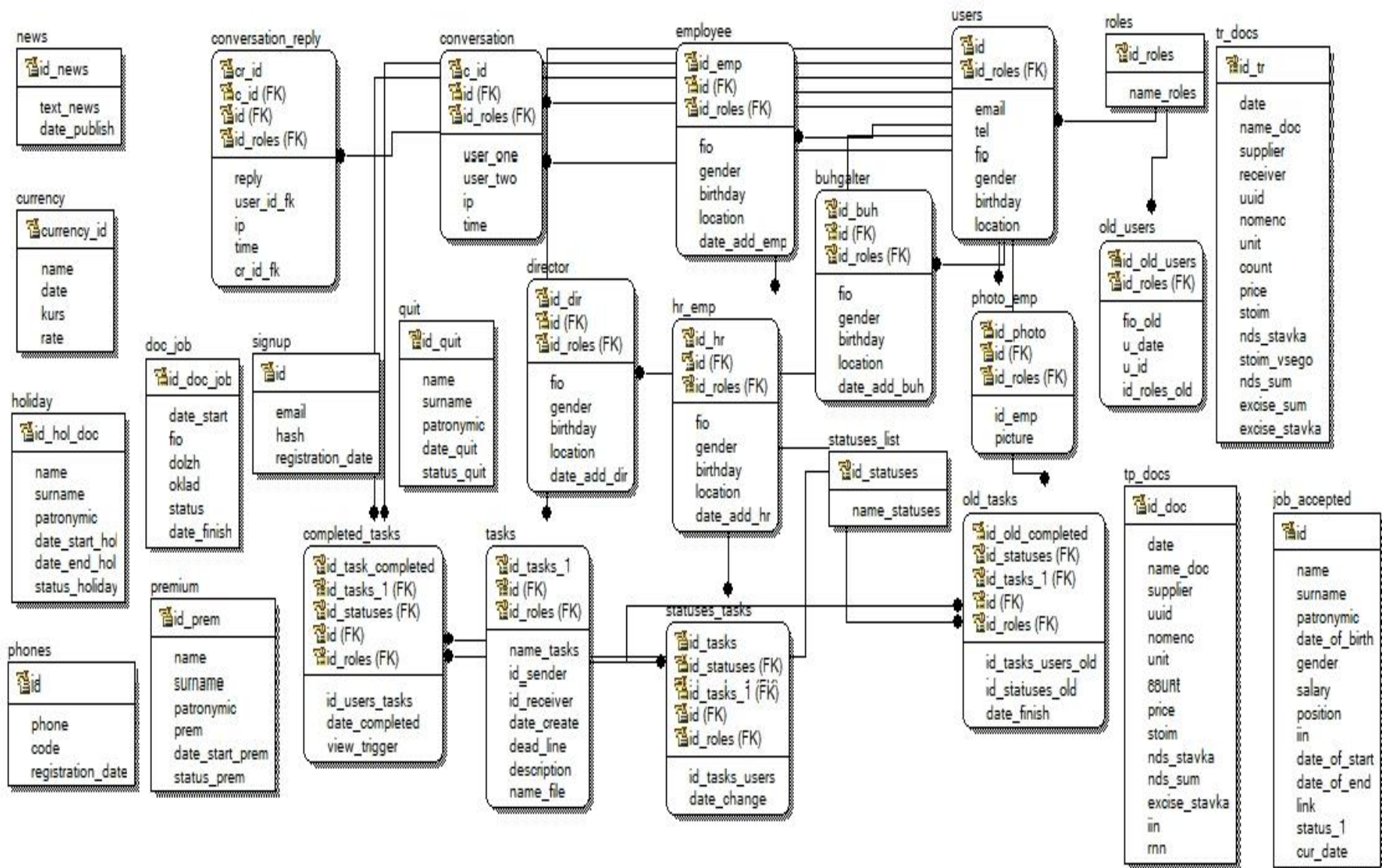


Рисунок 2.1 – Физическая модель данных

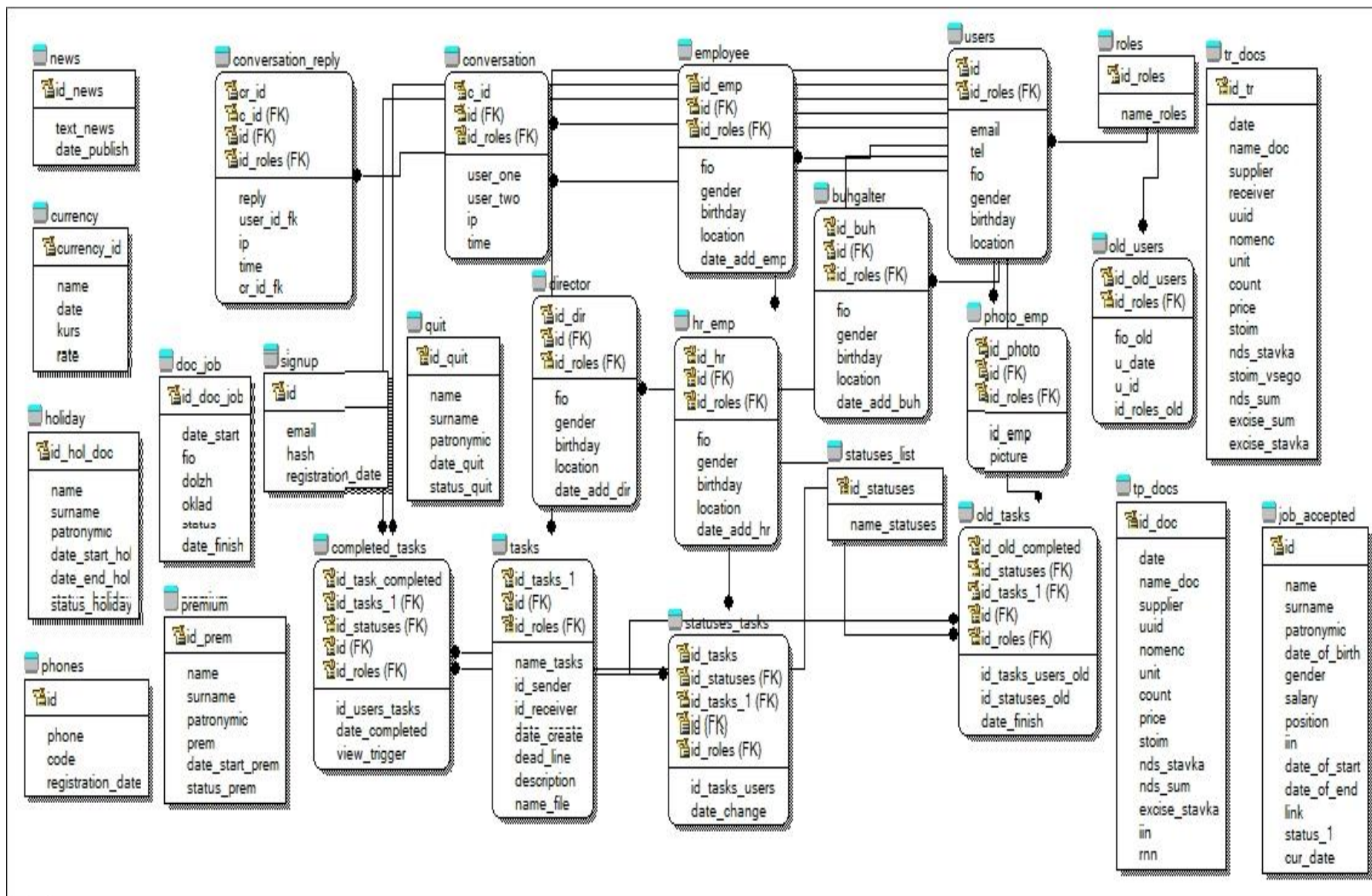


Рисунок 2.2 – Логическая модель данных

2.3 UML диаграммы базы данных

UML диаграмма классов базы данных была создана в программе IBM Rational Rose XDE.

Программа Rational Rose представляет собой CASE средство, использующееся для проектирования и разработки информационных систем, а также в качестве программного обеспечения для управления бизнес-процессами различными предприятиями. Наряду с другими CASE средствами (например, ARIS, BPwin или же ERwin) он также применяется для анализа и моделирования бизнес процессов. Первая версия данной утилиты была выпущена независимой компанией Rational Software, а в дальнейшем Rational Rose был куплен компанией-гигантом IBM [7].

Основное отличие программы Rational Rose от других CASE-средств заключается в объектно-ориентированном подходе. Графические модели, создаваемые с помощью этого CASE-средства, основаны на языке UML (Unified Modeling Language). За счет использования различных аспектов Rational Rose предоставляет юзерам (бизнес-аналитикам, инженерам баз данных, техническим специалистам и руководителям многих компаний) возможность создавать, анализировать, изменять и управлять различными моделями, используя единый язык моделирования, а также единый объектно-ориентированный подход [8].

Rational Rose имеет следующие возможности, благодаря которым создается моделирование бизнес процессов:

- поддержка объектного моделирования. Использование принципов объектного моделирования и языка UML способствует приближению модели процессов к основным требованиям бизнеса, а также упрощает вид их моделей;
- структурирование элементов. Модели процессов и их элементы могут быть изображены в виде графической структуры модели данных, которая наглядно отображает их состав и взаимодействия между собой;
- поддержка интеграции моделей. Благодаря использованию единого языка UML, утилита Rational Rose способствует объединению моделей бизнес процессов, а также моделей данных и приложений;
- интеграция с программными продуктами. Для расширения возможностей моделирования и анализа бизнес процессов в Rational Rose реализована возможность интеграции с другими программными продуктами, например, Microsoft Visual Studio;
- открытая архитектура. Она используется для дополнения существующего инструментария утилиты новыми возможностями и функциями.
- реверс проектирование. Эта особенность позволяет построить понятийную модель на основе уже имеющегося программного кода [9].

На рисунках 2.3 и 2.4 представлены UML диаграмма классов базы данных и UML диаграмма вариантов использования соответственно.

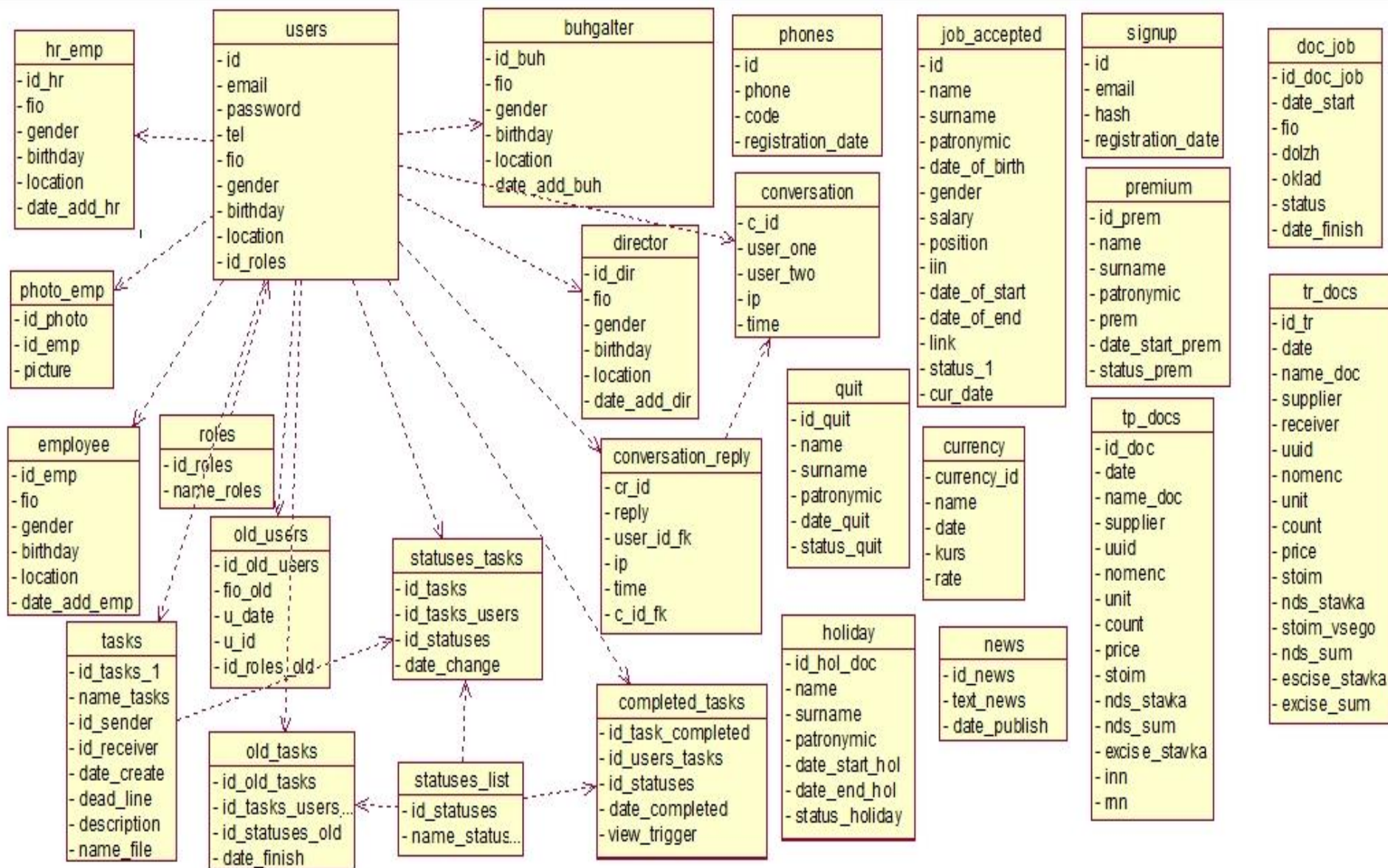


Рисунок 2.3 – UML диаграмма классов

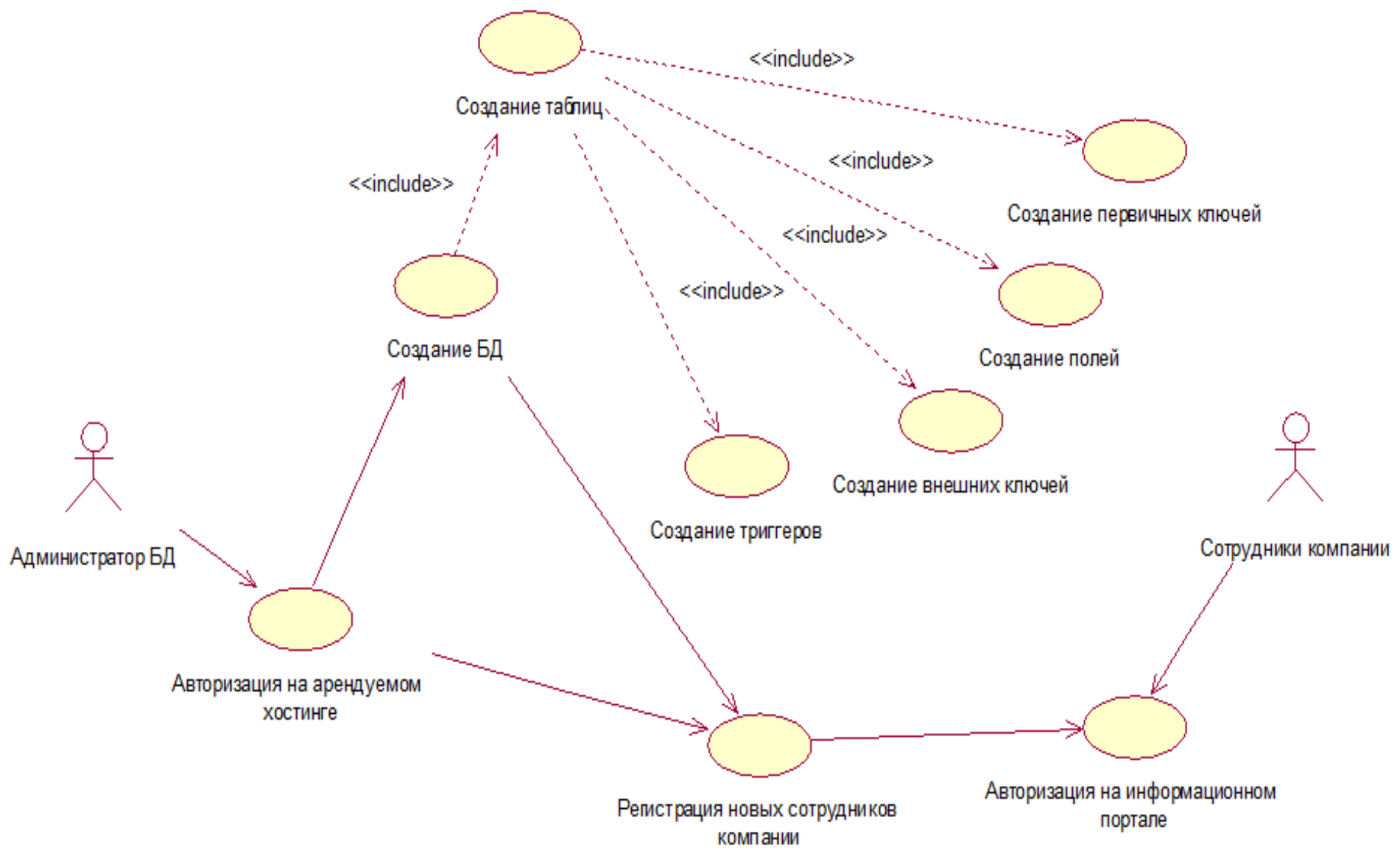


Рисунок 2.4 – UML диаграмма вариантов использования

3 Экспериментальная часть

3.1 Проектирование базы данных

Для информационного портала была спроектирована база данных на хостинговом сервере MySQL. На данный момент она содержит в себе 26 таблиц.

Для хранения данных сотрудников в БД используются следующие таблицы: «users» (id – primary key, email, password, tel, fio, gender, birthday, location, id_roles), «roles» (id_roles – primary key, name_roles), «buhgalter» (id_buh – primary key, fio, gender, birthday, location, date_add_buh), «director» (id_dir – primary key, fio, gender, birthday, location, date_add_dir), «hr_emp» (id_hr – primary key, fio, gender, birthday, location, date_add_hr) и «employee» (id – primary key, fio, gender, birthday, location, date_add_emp).

Таблицы «phones» (id – primary key, phone, code, registration_date) и «sighup» (id – primary key, email, hash, registration_date) содержат данные сотрудников, которые зарегистрировались в информационном портале по коду из СМС и по ссылке из электронного письма соответственно.

Таблица «news» (id_news – primary key, text_news, date_publish) хранит архив новостей портала.

В таблице «currency» (currency_id – primary key, name, date, kurs, rate) содержатся актуальные курсы разных валют.

Для реализации внутреннего чата были созданы 2 таблицы: «conversation» (c_id – primary key, user_one, user_two, ip, time) и «conversation_reply» (cr_id – primary key, reply, user_id_fk, ip, time, c_id_fk). В таблице «conversation» хранятся диалоги между сотрудниками, а в «conversation_reply» сообщения сотрудников.

В таблицах «tp_docs» (id_doc – primary key, date, name_doc, supplier, uuid, nomenc, unit, count, price, stoim, nds_stavka, nds_sum, excise_stavka, inn, rnn) и «tr_docs» (id_tr – primary key, date, name_doc, supplier, receiver, uuid, nomenc, unit, count, price, stoim, nds_stavka, stoim_vsego, nds_sum, excise_stavka, excise_sum) хранятся данные, которых берутся из документов 1С с помощью парсера.

В таблице «job_accepted» (id – primary key, name, surname, patronymic, date_of_birth_gender_salary, position, iin, date_of_start, date_of_end, link, status_1, cur_date) хранятся документы, связанные с принятием на работу новых лиц.

В таблице «quit» (id_quit – primary key, name, surname, patronymic, date_quit, status_quit) содержатся данные на увольнение сотрудников.

В таблицах «holiday» и «premium» находятся данные об отпуске и получении премий сотрудников соответственно.

Таблицы «tasks», «statuses_tasks», «statuses_list», «completed_tasks» и «old_tasks» содержат статусы задач, данные поставленных задач, выполненных задач, просроченных задач и данные, используемые для подсчета KPI.

Ниже приведен рисунок схемы проектирования базы данных (рисунок 3.1):

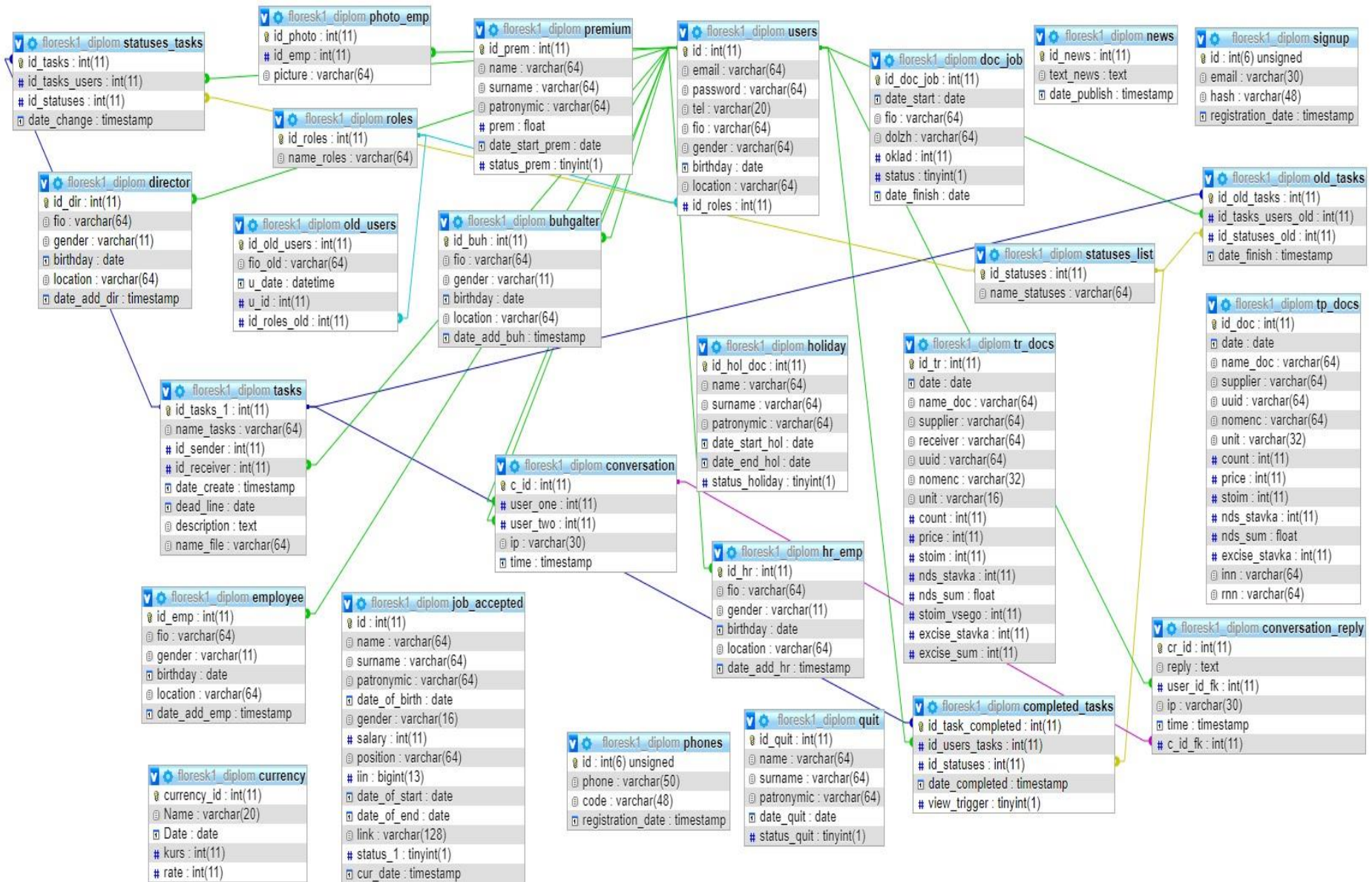


Рисунок 3.1 – Схема проектирования БД

3.2 Программная реализация базы данных

На основе реляционной модели произведена программная реализация. База данных содержит 26 таблиц (рисунок 3.2).

Таблица	Действие	Строки	Тип	Сравнение	Размер	Фрагментировано
buhgalter	★ Обзор Структура Поиск Вставить Очистить Удалить	2	InnoDB	utf8_general_ci	16 Киб	-
completed_tasks	★ Обзор Структура Поиск Вставить Очистить Удалить	0	InnoDB	utf8_general_ci	48 Киб	-
conversation	★ Обзор Структура Поиск Вставить Очистить Удалить	1	InnoDB	utf8_general_ci	48 Киб	-
conversation_reply	★ Обзор Структура Поиск Вставить Очистить Удалить	111	InnoDB	utf8_general_ci	48 Киб	-
currency	★ Обзор Структура Поиск Вставить Очистить Удалить	49	InnoDB	utf8_general_ci	16 Киб	-
director	★ Обзор Структура Поиск Вставить Очистить Удалить	2	InnoDB	utf8_general_ci	16 Киб	-
doc_job	★ Обзор Структура Поиск Вставить Очистить Удалить	0	InnoDB	utf8_general_ci	16 Киб	-
dokumentoobrot	★ Обзор Структура Поиск Вставить Очистить Удалить	0	InnoDB	utf8_general_ci	16 Киб	-
employee	★ Обзор Структура Поиск Вставить Очистить Удалить	2	InnoDB	utf8_general_ci	16 Киб	-
holiday	★ Обзор Структура Поиск Вставить Очистить Удалить	3	InnoDB	utf8_general_ci	16 Киб	-
hr_emp	★ Обзор Структура Поиск Вставить Очистить Удалить	2	InnoDB	utf8_general_ci	16 Киб	-
job_accepted	★ Обзор Структура Поиск Вставить Очистить Удалить	4	InnoDB	utf8_general_ci	16 Киб	-
manager	★ Обзор Структура Поиск Вставить Очистить Удалить	0	InnoDB	utf8_general_ci	16 Киб	-
news	★ Обзор Структура Поиск Вставить Очистить Удалить	0	InnoDB	utf8_general_ci	16 Киб	-
old_tasks	★ Обзор Структура Поиск Вставить Очистить Удалить	1	InnoDB	utf8_general_ci	48 Киб	-
old_users	★ Обзор Структура Поиск Вставить Очистить Удалить	2	InnoDB	utf8_general_ci	32 Киб	-
phones	★ Обзор Структура Поиск Вставить Очистить Удалить	5	InnoDB	utf8_general_ci	16 Киб	-
photo_emp	★ Обзор Структура Поиск Вставить Очистить Удалить	2	InnoDB	utf8_general_ci	32 Киб	-
premium	★ Обзор Структура Поиск Вставить Очистить Удалить	1	InnoDB	utf8_general_ci	16 Киб	-
quit	★ Обзор Структура Поиск Вставить Очистить Удалить	1	InnoDB	utf8_general_ci	16 Киб	-
roles	★ Обзор Структура Поиск Вставить Очистить Удалить	4	InnoDB	utf8_general_ci	16 Киб	-
signup	★ Обзор Структура Поиск Вставить Очистить Удалить	2	InnoDB	utf8_general_ci	16 Киб	-
statuses_list	★ Обзор Структура Поиск Вставить Очистить Удалить	7	InnoDB	utf8_general_ci	16 Киб	-
statuses_tasks	★ Обзор Структура Поиск Вставить Очистить Удалить	9	InnoDB	utf8_general_ci	48 Киб	-
tasks	★ Обзор Структура Поиск Вставить Очистить Удалить	10	InnoDB	utf8_general_ci	32 Киб	-
tp_docs	★ Обзор Структура Поиск Вставить Очистить Удалить	2	InnoDB	utf8_general_ci	16 Киб	-
tr_docs	★ Обзор Структура Поиск Вставить Очистить Удалить	10	InnoDB	utf8_general_ci	16 Киб	-
users	★ Обзор Структура Поиск Вставить Очистить Удалить	15	InnoDB	utf8_general_ci	32 Киб	-
28 таблиц	Всего	247	InnoDB	utf8_general_ci	672 Киб	0 Байт

Рисунок 3.2 – Список таблиц БД

Само создание базы происходило на хостинге (блок рисунков 3.3):

Добавить базу данных

Общие

Имя базы данных *

Сервер баз данных

Соответствующий сайт

Пользователи

Создайте пользователя базы данных по умолчанию. Это пользователь, от лица которого Plesk будет подключаться к базе данных. Если базе данных не назначить ни одного пользователя, она будет недоступна.

Создать пользователя базы данных

Имя пользователя базы данных *

Новый пароль *

Подтверждение пароля *

Пользователь имеет доступ ко всем базам данных выбранной подписки

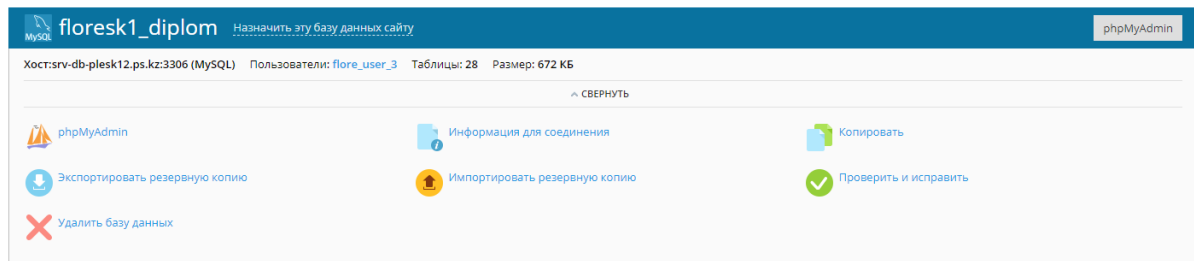
Управление доступом

Разрешить только локальные подключения

Разрешить удаленные подключения с любого компьютера

Разрешить удаленные подключения с указанных компьютеров

* Обязательные поля



Блок рисунков 3.3 – Создание базы данных

3.4. Создание таблиц users, roles и tasks продемонстрированы в блоке рисунков

```

CREATE TABLE `users` (
  `id` int(11) NOT NULL,
  `email` varchar(64) NOT NULL,
  `password` varchar(64) NOT NULL,
  `tel` varchar(20) NOT NULL,
  `fio` varchar(64) NOT NULL,
  `gender` varchar(64) NOT NULL,
  `birthday` date NOT NULL,
  `location` varchar(64) NOT NULL,
  `id_roles` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
ALTER TABLE `users`
  ADD PRIMARY KEY (`id`),
  ADD KEY `PK10` (`id_roles`);
ALTER TABLE `users`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=22;
ALTER TABLE `users`
  ADD CONSTRAINT `PK10` FOREIGN KEY (`id_roles`) REFERENCES `roles` (`id_roles`);
COMMIT;

CREATE TABLE `roles` (
  `id_roles` int(11) NOT NULL,
  `name_roles` varchar(64) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
ALTER TABLE `roles`
  ADD PRIMARY KEY (`id_roles`);
COMMIT;

CREATE TABLE `tasks` (
  `id_tasks_1` int(11) NOT NULL,
  `name_tasks` varchar(64) NOT NULL,
  `id_sender` int(11) NOT NULL,
  `id_receiver` int(11) NOT NULL,
  `date_create` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE current_timestamp(),
  `dead_line` date NOT NULL,
  `description` text NOT NULL,
  `name_file` varchar(64) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
ALTER TABLE `tasks`
  ADD PRIMARY KEY (`id_tasks_1`),
  ADD KEY `PK14` (`id_receiver`);
ALTER TABLE `tasks`
  MODIFY `id_tasks_1` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=20;
ALTER TABLE `tasks`
  ADD CONSTRAINT `PK14` FOREIGN KEY (`id_receiver`) REFERENCES `users` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;
COMMIT;

```

Блок рисунков 3.4 – Создание таблиц

3.5. Заполнение вышеперечисленных страниц представлено в блоке рисунков

```
INSERT INTO `users` (`id`, `email`, `password`, `tel`, `fio`, `gender`, `birthday`, `location`, `id_roles`) VALUES
(4, 'sofiasan_1998@mail.ru', 'sss', '87025477149', 'San Sofiya Archuanovna', 'F', '0000-00-00', '', 1),
(5, 'garold_pain@garold.com', '123', '+7775553535', 'Гарольд, Скрывающий Воль', 'M', '2020-02-02', 'Алматы', 2),
(6, 'почта', 'йцу', '+77783245699', 'Рикардо Милос', 'M', '2019-11-05', 'Almaty', 3),
(7, '123', '123', '+77023385870', 'Марина Ивановна', 'Ж', '2020-02-03', 'Алматы', 4),
(8, 'sungsofiya@gmail.com', '123', '87476874337', 'San Sofiya', 'F', '0000-00-00', 'Almaty', 1),
(9, 'qwe', 'qwe', '+77783245699', 'qwe', 'qwe', '2020-03-27', 'qwe', 1),
(12, 'wwe', 'wewe', '+77783245699', 'ewew', 'wewe', '2020-03-27', 'ewewe', 2),
(13, 'hr', 'hr', '+77783245699', 'hr', 'hr', '2020-03-25', 'hr', 2),
(14, 'buh', 'buh', 'buh', 'buh', 'buh', '2020-03-27', 'buh', 3),
(15, 'em', 'em', 'em', 'em', 'em', '2020-03-26', 'em', 4),
(16, 'qwe', 'qwe', 'qwe', 'qwe', 'qwe', '2020-03-25', 'qwe', 1),
(18, '123@mail.ru', '123', '123', '123', 'M', '1990-03-09', '123', 1),
(19, '234@mail.ru', '234', '234', '234', 'M', '1990-03-09', '234', 2),
(20, '345@mail.ru', '345', '345', '345', 'M', '1990-03-09', '345', 3),
(21, '456@mail.ru', '456', '456', '456', 'M', '1990-03-09', '456', 4);

INSERT INTO `roles` (`id_roles`, `name_roles`) VALUES
(1, 'Директор'),
(2, 'Кадровик'),
(3, 'Бухгалтер'),
(4, 'Сотрудник');

INSERT INTO `tasks` (`id_tasks_1`, `name_tasks`, `id_sender`, `id_receiver`, `date_create`, `dead_line`, `description`, `name_file`) VALUES
(4, 'asd', 1, 8, '2020-03-22 20:35:15', '2020-03-22', 'qqq', 'aaa'),
(5, 'qwe', 1, 7, '2020-03-22 20:35:15', '2020-03-16', 'aaaaaq', 'qqqqqq'),
(6, 'aaa', 1, 5, '2020-03-22 20:37:30', '2020-03-18', 'aaaa', 'aaaa'),
(7, 'bbbb', 1, 7, '2020-03-22 20:37:30', '2020-03-17', 'bbbbbb', 'bbbbbb'),
(10, 'sma', 1, 4, '2020-03-23 14:47:34', '2020-05-05', '20202', '2'),
(11, '23/03/2020', 1, 4, '2020-03-23 18:02:04', '2020-03-28', 'adil', '1'),
(12, 'advvadfvadvf', 1, 5, '2020-03-23 18:05:16', '2020-03-13', 'pasha', ''),
(13, 'sdf', 1, 4, '2020-03-23 18:17:33', '2020-05-08', '2558967', 'name'),
(14, 'asdf', 1, 4, '2020-03-23 18:18:36', '2020-02-20', '20202020202020', './httpdocs/docx_dir/1234.txt'),
(15, 'asd', 1, 4, '2020-03-24 14:10:41', '2020-03-22', 'aaa', 'aaa');
```

Блок рисунков 3.5 – Заполнение таблиц

Пример выполнения запроса для вывода информации о сотрудниках, занимающих должность «Бухгалтер» (рисунок 3.6).

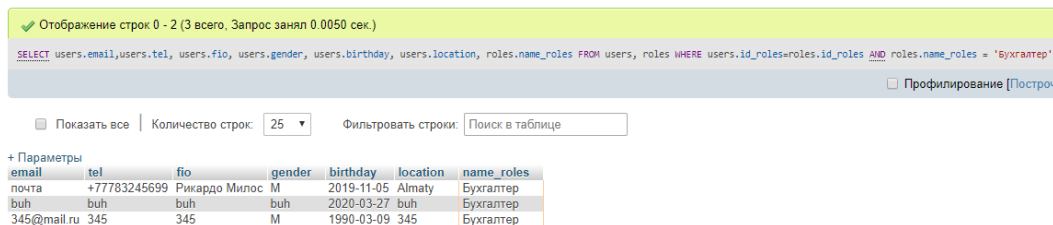


Рисунок 3.6 – Пример выполнения запроса

Пример запроса, выполняющего вывод всех задач со статусом «1» у сотрудника с id = 4 (рисунок 3.7).

```
SELECT * FROM tasks, statuses_tasks WHERE tasks.id_tasks_1 = statuses_tasks.id_tasks AND tasks.id_receiver = statuses_tasks.id_tasks_users AND statuses_tasks.id_statuses = '1' AND tasks.id_receiver = '4'
```

Профилирование Построчное редактирование

Показать все | Количество строк: 25 | Фильтровать строки: Поиск в таблице

+ Параметры

id_tasks_1	name_tasks	id_sender	id_receiver	date_create	dead_line	description	name_file	id_tasks	id_tasks_users	id_statuses	date_change
10	ыва	1	4	2020-03-23 20:47:34	2020-05-05	20202	2	10	4	1	2020-03-23 20:47:34
11	23/03/2020	1	4	2020-03-24 00:02:04	2020-03-28	adil	1	11	4	1	2020-03-24 00:02:04
13	sdf	1	4	2020-03-24 00:17:33	2020-05-08	2568967	name	13	4	1	2020-03-24 00:17:33
14	asdf	1	4	2020-03-24 00:18:36	2020-02-20	202020202020	../httpdocs/docx_dir/1234.txt	14	4	1	2020-03-24 00:18:36

Рисунок 3.7 - Пример выполнения запроса

Создадим и проверим триггер для таблицы users, который при добавлении нового сотрудника будет заносить его в таблицу в зависимости от его роли. Работоспособность триггера представлена в блоке рисунков 3.8.

Столбец	Тип	Функция	Null	Значение
id	int(11)	<input type="text"/>	<input type="checkbox"/>	<input type="text"/>
email	varchar(64)	<input type="text"/>	<input type="checkbox"/>	abc@abc.ru
password	varchar(64)	<input type="text"/>	<input type="checkbox"/>	qwerty
tel	varchar(20)	<input type="text"/>	<input type="checkbox"/>	87025477149
fio	varchar(64)	<input type="text"/>	<input type="checkbox"/>	Иван Иванов
gender	varchar(64)	<input type="text"/>	<input type="checkbox"/>	M
birthday	date	<input type="text"/>	<input type="checkbox"/>	1976-04-14
location	varchar(64)	<input type="text"/>	<input type="checkbox"/>	Алматы
id_roles	int(11)	<input type="text"/>	<input type="checkbox"/>	Бухгалтер - 3

[Вперёд](#)

id	email	password	tel	fio	gender	birthday	location	id_roles
4	sofiasan_1998@mail.ru	sss	87025477149	San Sofiya Archuanovna	F	0000-00-00		1
5	garold_pain@garold.com	123	+7775553535	Гарольд, Скрывающий Боль	M	2020-02-02	Алматы	2
6	почта	йцу	+77783245699	Рикардо Милос	M	2019-11-05	Алматы	3
7	123	123	+77023385870	Марина Ивановна	Ж	2020-02-03	Алматы	4
8	sungsofiya@gmail.com	123	87476874337	San Sofiya	F	0000-00-00	Алматы	1
9	qwe	qwe	+77783245699	qwe	qwe	2020-03-27	qwe	1
12	wwe	wewe	+77783245699	ewew	wewe	2020-03-27	ewewe	2
13	hr	hr	+77783245699	hr	hr	2020-03-25	hr	2
14	buh	buh	buh	buh	buh	2020-03-27	buh	3
15	em	em	em	em	em	2020-03-26	em	4
16	qwe	qwe	qwe	qwe	qwe	2020-03-25	qwe	1
18	123@mail.ru	123	123	123	M	1990-03-09	123	1
19	234@mail.ru	234	234	234	M	1990-03-09	234	2
20	345@mail.ru	345	345	345	M	1990-03-09	345	3
21	456@mail.ru	456	456	456	M	1990-03-09	456	4
22	abc@abc.ru	qwerty	87025477149	Иван Иванов	M	1976-04-14	Алматы	3

id_buh	fio	gender	birthday	location	date_add_buh
14	buh	buh	2020-03-27	buh	2020-03-27 11:13:33
20	345	345	1990-03-09	345	2020-03-29 17:42:28
22	Иван Иванов	M	1976-04-14	Алматы	2020-04-02 10:17:14

Блок рисунков 3.8 – Работоспособность триггера

Как видно из рисунков, приведенных выше, триггер сработал на добавление нового сотрудника и определил его в таблицу «buhgalter», т.к. должность его была «бухгалтер».

Затем, создадим и проверим триггер для таблицы users, который при добавлении нового сотрудника будет заносить его в таблицу photo_emp в которой хранятся фото сотрудников. Работоспособность триггера представлена в блоке рисунков 3.9.

Столбец	Тип	Функция	Null	Значение
id	int(11)			
email	varchar(64)			example@example.com
password	varchar(64)			123
tel	varchar(20)			87025477149
name	varchar(64)			Василий
surname	varchar(64)			Федоров
patronymic	varchar(64)			Васильевич
gender	varchar(64)			М
birthday	date			1977-04-08
location	varchar(64)			Алматы
id_roles	int(11)			Бухгалтер - 3

	id	email	password	tel	name	surname	patronymic	gender	birthday	location	id_roles
<input type="checkbox"/>	4	sofiasan_1996@mail.ru	sss	87025477149	Sofiya	San	Archuanovna	F	0000-00-00		1
<input type="checkbox"/>	5	garold_rain@garold.com	123	+7775553535	Гарольд	Иванов	Аржадьевич	M	2020-02-02	Алматы	2
<input type="checkbox"/>	6	example@example.com	йцу	+77783245699				M	2019-11-05	Алматы	3
<input type="checkbox"/>	7	example1@example.com	123	+77023385870				Ж	2020-02-03	Алматы	4
<input type="checkbox"/>	8	sungsofiya@gmail.com	123	87476874337	Sofiya	Sung	Arche	F	0000-00-00	Алматы	1
<input type="checkbox"/>	9	example2@example.com	qwe	+77783245699				qwe	2020-03-27	qwe	1
<input type="checkbox"/>	12	example3@example.com	wewe	+77783245699				wewe	2020-03-27	ewewe	2
<input type="checkbox"/>	13	example4@example.com	hr	+77783245699				hr	2020-03-25	hr	2
<input type="checkbox"/>	14	example5@example.com	buh	+77783245696				buh	2020-03-27	buh	3
<input type="checkbox"/>	15	example6@example.com	em	+77783245691				em	2020-03-26	em	4
<input type="checkbox"/>	16	example7@example.com	qwe	+77783245699	Маррк	Цукерберг	Маркович	qwe	2020-03-25	qwe	1
<input type="checkbox"/>	18	123@mail.ru	123	+77783232789	Адиль	Алдамжаров	Хайратович	Мужчина	1999-03-09	123	1
<input type="checkbox"/>	19	234@mail.ru	234	+77783212569	Владислав	Тен	Владимирович	M	1990-03-09	234	2
<input type="checkbox"/>	20	345@mail.ru	345	+77783245669	Владислав	Тен	Владимирович	M	1990-03-09	345	3
<input type="checkbox"/>	21	456@mail.ru	456	+77783245199	Владислав	Тен	Владимирович	M	1990-03-09	456	4
<input type="checkbox"/>	22	567@abc.ru	567	87025477149	Владислав	Тен	Владимирович	M	1976-04-14	Алматы	1
<input type="checkbox"/>	26	example@example.com	123	87025477149	Василий	Федоров	Васильевич	M	1977-04-08	Алматы	3

	id_photo	id_emp	picture
<input type="checkbox"/>	1	5	images_emp/test.jpg
<input type="checkbox"/>	2	6	images_emp/test_1.jpg
<input type="checkbox"/>	5	18	images_emp/_18_chuzheplanetnyy_zamok.jpg
<input type="checkbox"/>	10	16	images_emp/_16_bad_boy.jpg
<input type="checkbox"/>	11	19	images_emp/test_1.jpg
<input type="checkbox"/>	12	4	images_emp/_4_hamster.jpg
<input type="checkbox"/>	13	26	images_emp/no-avatar-png.png

Блок рисунков 3.9 – Работоспособность триггера

После этого, создаем триггер, который будет при создании новой задачи присваивать ей статус «Новая». Работоспособность триггера представлена в блоке рисунков 3.10.

← T →	id_tasks_1	name_tasks	id_sender	id_receiver	date_create	dead_line	description	name_file
Изменить Копировать Удалить	10	ыва	8	4	2020-04-04 21:44:51	2020-05-05	20202	2
Изменить Копировать Удалить	11	23/03/2020	9	4	2020-04-04 21:44:54	2020-03-28	adil	1
Изменить Копировать Удалить	12	adfvadvdf	12	5	2020-04-04 21:45:07	2020-03-13	pasha	
Изменить Копировать Удалить	13	sdf	13	4	2020-04-04 21:45:11	2020-05-08	2558967	name
Изменить Копировать Удалить	27	SVAP	14	4	2020-04-04 21:45:14	2020-04-18	3333	../httpdocs/docx_dir/AA_v3.log
Изменить Копировать Удалить	30	qwe	12	20	2020-04-05 23:51:20	2020-03-18	qwe	qwe
Изменить Копировать Удалить	34	Vladislav	16	18	2020-04-05 23:59:09	2020-04-19	54555	../httpdocs/docx_dir/
Изменить Копировать Удалить	35	test	5	15	2020-04-06 23:27:03	2020-04-17	ункуцунц	../httpdocs/docx_dir/
Изменить Копировать Удалить	36	тест	21	5	2020-04-07 22:01:05	2020-04-14	йцц	../httpdocs/docx_dir/AMMY_service.log
Изменить Копировать Удалить	38	фчфч	7	5	2020-04-07 00:58:28	2020-03-18	фыфыф	Нет файла
Изменить Копировать Удалить	39	ячяся	12	5	2020-04-07 00:59:50	2020-04-08	ййцй	Нет файла
Изменить Копировать Удалить	42	qwe	22	20	2020-04-07 21:43:46	2020-03-18	qqw	Нет файла
Изменить Копировать Удалить	43	Vladislav	16	4	2020-04-07 21:43:48	2020-04-22	123	../httpdocs/docx_dir/AA_v3.log
Изменить Копировать Удалить	45	Vladislav	16	4	2020-04-07 21:47:49	2020-04-25	55555	Нет файла
Изменить Копировать Удалить	46	Vladislav	16	4	2020-04-07 21:48:16	2020-04-30	55555	../httpdocs/docx_dir/AMMY_service.log
Изменить Копировать Удалить	47	TEST888	19	5	2020-04-08 22:55:15	2020-04-25	8888	Нет файла
Изменить Копировать Удалить	48	Тест	5	19	2020-04-08 22:58:33	2020-04-12	Картинка	../httpdocs/docx_dir/logo.png
Изменить Копировать Удалить	49	Vladislav	19	5	2020-04-08 23:09:15	2020-04-25	999	../httpdocs/docx_dir/111.mxl
Изменить Копировать Удалить	50	Тест	5	19	2020-04-08 23:36:46	2020-04-09	ке7ноубн	Нет файла
Изменить Копировать Удалить	51	Тест	5	19	2020-04-08 23:40:45	2020-04-23	sdsq	../httpdocs/docx_dir/logo.png
Изменить Копировать Удалить	52	Не страдать	18	5	2020-04-09 01:04:01	2020-12-31	Не страдай	Нет файла
Изменить Копировать Удалить	53	Пример названия	14	26	2020-04-27 10:39:21	2020-04-30	Пример описания	Нет файла

← T →	id_tasks	id_tasks_users	id_tasks_sender	id_statuses	date_change
Изменить Копировать Удалить	13	4	13	1	2020-03-24 00:17:33
Изменить Копировать Удалить	46	4	16	1	2020-04-07 21:48:16
Изменить Копировать Удалить	52	5	18	1	2020-04-09 01:04:01
Изменить Копировать Удалить	53	26	14	1	2020-04-27 10:39:21

↑ Отметить все *С отмеченными* Изменить Копировать Удалить Экспорт Новая

Блок рисунков 3.10 – Работоспособность триггера

Также были созданы триггеры, которые срабатывают на изменение статуса задач, например, при изменении статуса на статус «На рассмотрении», эта задача отправляется тому, кто ее поставил. Если он ее отклонит, то эта задача вернется к получателю со статусом «Отклонена», а если отправитель ее примет, то статус изменится на «Принята» и задача отправится в таблицу выполненных и просроченных задач old_tasks. Работоспособность вышеназванных триггеров представлена в блоке рисунков 3.11.

← T →	id_tasks	id_tasks_users	id_tasks_sender	id_statuses	date_change
Изменить Копировать Удалить	13	4	13	1	2020-03-24 00:17:33
Изменить Копировать Удалить	46	4	16	1	2020-04-07 21:48:16
Изменить Копировать Удалить	52	5	18	1	2020-04-09 01:04:01
Изменить Копировать Удалить	53	26	14	6	2020-04-27 10:46:12

		id_task_completed	id_users_tasks	id_tasks_sender	id_statuses	date_completed	view_trigger
<input type="checkbox"/>	Изменить Копировать Удалить	51	19	5	5	2020-04-09 00:00:08	1
<input type="checkbox"/>	Изменить Копировать Удалить	53	26	14	6	2020-04-27 10:48:04	1

		id_tasks	id_tasks_users	id_tasks_sender	id_statuses	date_change
<input type="checkbox"/>	Изменить Копировать Удалить	13	4	13	1	2020-03-24 00:17:33
<input type="checkbox"/>	Изменить Копировать Удалить	46	4	16	1	2020-04-07 21:48:16
<input type="checkbox"/>	Изменить Копировать Удалить	52	5	18	1	2020-04-09 01:04:01
<input type="checkbox"/>	Изменить Копировать Удалить	53	26	14	5	2020-04-27 10:48:42

		id_old_tasks	id_tasks_users_old	id_tasks_sender_old	id_statuses_old	date_finish
<input type="checkbox"/>	Изменить Копировать Удалить	10	4	8	3	2020-04-07 00:02:05
<input type="checkbox"/>	Изменить Копировать Удалить	11	4	9	4	2020-04-06 23:15:13
<input type="checkbox"/>	Изменить Копировать Удалить	12	5	12	4	2020-04-06 23:15:13
<input type="checkbox"/>	Изменить Копировать Удалить	27	4	14	4	2020-04-26 21:45:45
<input type="checkbox"/>	Изменить Копировать Удалить	30	20	12	4	2020-04-06 23:15:13
<input type="checkbox"/>	Изменить Копировать Удалить	34	18	16	4	2020-04-26 21:45:45
<input type="checkbox"/>	Изменить Копировать Удалить	35	15	5	3	2020-04-08 23:31:57
<input type="checkbox"/>	Изменить Копировать Удалить	36	5	21	4	2020-04-26 21:45:45
<input type="checkbox"/>	Изменить Копировать Удалить	38	5	7	4	2020-04-07 00:58:32
<input type="checkbox"/>	Изменить Копировать Удалить	39	5	12	4	2020-04-09 00:55:49
<input type="checkbox"/>	Изменить Копировать Удалить	42	19	22	4	2020-04-08 23:34:33
<input type="checkbox"/>	Изменить Копировать Удалить	43	4	16	4	2020-04-26 21:45:45
<input type="checkbox"/>	Изменить Копировать Удалить	45	4	16	4	2020-04-26 21:45:45
<input type="checkbox"/>	Изменить Копировать Удалить	47	5	19	3	2020-04-08 22:56:16
<input type="checkbox"/>	Изменить Копировать Удалить	48	19	5	3	2020-04-08 22:59:59
<input type="checkbox"/>	Изменить Копировать Удалить	49	5	19	3	2020-04-08 23:22:48
<input type="checkbox"/>	Изменить Копировать Удалить	50	19	5	4	2020-04-10 00:34:06
<input type="checkbox"/>	Изменить Копировать Удалить	51	19	5	4	2020-04-26 21:45:45
<input type="checkbox"/>	Изменить Копировать Удалить	53	26	14	3	2020-04-27 10:49:27

Блок рисунков 3.11 – Работоспособность триггеров

Далее, рассмотрим триггер, который при добавлении нового сотрудника добавляет ему стандартный аватар, который потом можно будет заменить на сайте. Работа данного триггера и стандартный аватар представлены в блоке рисунков 3.12.

		id	email	password	tel	name	surname	patronymic	gender	birthday	location	id_roles
<input type="checkbox"/>	Изменить Копировать Удалить	4	sofiasan_1998@mail.ru	sss	87025477149	Sofiya	San	Archuanovna	F	0000-00-00		1
<input type="checkbox"/>	Изменить Копировать Удалить	5	garold_pain@garold.com	123	+7775553535	Гарольд	Иванов	Аркадьевич	M	2020-02-02	Алматы	2
<input type="checkbox"/>	Изменить Копировать Удалить	6	example@example.com	йцу	+77783245699				M	2019-11-05	Almaty	3
<input type="checkbox"/>	Изменить Копировать Удалить	7	example1@example.com	123	+77023385870				Ж	2020-02-03	Алматы	4
<input type="checkbox"/>	Изменить Копировать Удалить	8	sungsofiya@gmail.com	123	87476874337	Sofiya	Sung	Arche	F	0000-00-00	Almaty	1
<input type="checkbox"/>	Изменить Копировать Удалить	9	example2@example.com	qwe	+77783245699				qwe	2020-03-27	qwe	1
<input type="checkbox"/>	Изменить Копировать Удалить	12	example3@example.com	wewe	+77783245699				wewe	2020-03-27	ewewe	2
<input type="checkbox"/>	Изменить Копировать Удалить	13	example4@example.com	hr	+77783245699				hr	2020-03-25	hr	2
<input type="checkbox"/>	Изменить Копировать Удалить	14	example5@example.com	buh	+77783245696				buh	2020-03-27	buh	3
<input type="checkbox"/>	Изменить Копировать Удалить	15	example6@example.com	em	+77783245691				em	2020-03-26	em	4
<input type="checkbox"/>	Изменить Копировать Удалить	16	example7@example.com	qwe	+77883245699	Маррк	Цукерберг	Маркович	qwe	2020-03-25	qwe	1
<input type="checkbox"/>	Изменить Копировать Удалить	18	123@mail.ru	123	+77783232789	Адиль	Алдамжаров	Хайратович	Мужчина	1999-03-09	123	1
<input type="checkbox"/>	Изменить Копировать Удалить	19	234@mail.ru	234	+77783212569	Владислав	Тен	Владимирович	M	1990-03-09	234	2
<input type="checkbox"/>	Изменить Копировать Удалить	20	345@mail.ru	345	+77783245669	Владислав	Тен	Владимирович	M	1990-03-09	345	3
<input type="checkbox"/>	Изменить Копировать Удалить	21	456@mail.ru	456	+77783245199	Владислав	Тен	Владимирович	M	1990-03-09	456	4
<input type="checkbox"/>	Изменить Копировать Удалить	22	567@abc.ru	567	87025477149	Владислав	Тен	Владимирович	M	1976-04-14	Алматы	1
<input type="checkbox"/>	Изменить Копировать Удалить	26	example@example.com	123	87025477149	Василий	Федоров	Васильевич	M	1977-04-08	Алматы	3
<input type="checkbox"/>	Изменить Копировать Удалить	27	asdda	sssdcs	575675	wgs	gvsvdf	regerdgr	ergergr	1980-03-09	hrtyhrr	2
<input type="checkbox"/>	Изменить Копировать Удалить	28	qwe@qwe.com	qwe	87025477149	Иван	Федоров	Иванов	M	1989-04-08	Алматы	4

			id_photo	id_emp	picture				
<input type="checkbox"/>		Изменить		Копировать		Удалить	1	5	images_emp\test.jpg
<input type="checkbox"/>		Изменить		Копировать		Удалить	2	6	images_emp\test_1.jpg
<input type="checkbox"/>		Изменить		Копировать		Удалить	5	18	images_emp/_18_chuzheplanetnyy_zamok.jpg
<input type="checkbox"/>		Изменить		Копировать		Удалить	10	16	images_emp/_16_bad_boy.jpg
<input type="checkbox"/>		Изменить		Копировать		Удалить	11	19	images_emp\test_1.jpg
<input type="checkbox"/>		Изменить		Копировать		Удалить	12	4	images_emp/_4_hamster.jpg
<input type="checkbox"/>		Изменить		Копировать		Удалить	13	26	images_emp/no-avatar-png.png
<input type="checkbox"/>		Изменить		Копировать		Удалить	14	27	images_emp/no-avatar-png.png
<input type="checkbox"/>		Изменить		Копировать		Удалить	15	28	images_emp/no-avatar-png.png



Блок рисунков 3.12 – Работоспособность триггеров

Все разработанные триггеры хранятся в базе данных, а именно в специальном подразделе «Триггеры» на главном интерфейсе phrmyadmin и, которые, в дальнейшем, можно будет отредактировать, экспортировать или удалить (рисунок 3.13).

Триггеры						
	Имя	Таблица	Действие	Время	Событие	
<input type="checkbox"/>	create_status_task	tasks	Изменить Экспорт Удалить	AFTER	INSERT	
<input type="checkbox"/>	delete_active_task	completed_tasks	Изменить Экспорт Удалить	BEFORE	UPDATE	
<input type="checkbox"/>	delete_task	tasks	Изменить Экспорт Удалить	BEFORE	DELETE	
<input type="checkbox"/>	delete_users	users	Изменить Экспорт Удалить	AFTER	DELETE	
<input type="checkbox"/>	photo_add	users	Изменить Экспорт Удалить	AFTER	INSERT	
<input type="checkbox"/>	roles_add	users	Изменить Экспорт Удалить	AFTER	INSERT	
<input type="checkbox"/>	status_cancelled	completed_tasks	Изменить Экспорт Удалить	BEFORE	UPDATE	
<input type="checkbox"/>	update_active_task	statuses_tasks	Изменить Экспорт Удалить	AFTER	UPDATE	

Рисунок 3.13 – Список триггеров

Стоит отметить, что все загруженные фотографии в профили сотрудников компании хранятся не в самой базе данных, а в папке «images_emp», находящейся на хостинге, то есть в самой базе данных хранятся пути к этим самым фотографиям сотрудников (рисунок 3.14).

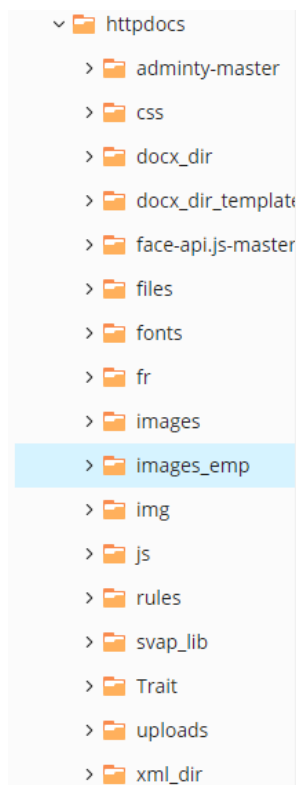


Рисунок 3.14 – Расположение папки «images_emp»

Содержимое папки «images_emp» и пути к фотографиям сотрудников, хранящиеся в самой базе данных в поле «picture» таблицы «photo_emp», приведены в блоке рисунков 3.15.

			id_photo	id_emp	picture	
<input type="checkbox"/>	Изменить	Копировать	Удалить	1	5	images_emp\test.jpg
<input type="checkbox"/>	Изменить	Копировать	Удалить	2	6	images_emp\test_1.jpg
<input type="checkbox"/>	Изменить	Копировать	Удалить	5	18	images_emp/_18_chuzheplanetnyy_zamok.jpg
<input type="checkbox"/>	Изменить	Копировать	Удалить	10	16	images_emp/_16_bad_boy.jpg
<input type="checkbox"/>	Изменить	Копировать	Удалить	11	19	images_emp\test_1.jpg
<input type="checkbox"/>	Изменить	Копировать	Удалить	12	4	images_emp/_4_hamster.jpg
<input type="checkbox"/>	Изменить	Копировать	Удалить	13	26	images_emp/no-avatar-png.png
<input type="checkbox"/>	Изменить	Копировать	Удалить	14	27	images_emp/no-avatar-png.png
<input type="checkbox"/>	Изменить	Копировать	Удалить	15	28	images_emp/no-avatar-png.png

Домашняя папка > httpdocs > images_emp >

Имя	Дата последнего изменения ↓	Размер	Права	Пользователь	Группа
..	28/04/2020 17:14		rwX r-x ---	floresk1	psaserv
_18_chuzheplanetnyy_zamok.jpg	29/04/2020 07:34	4.0 КБайт	rw- r-- r--	floresk1	psacln
_20_Z7tvcfjt7AU.jpg	9/04/2020 18:36	4.0 КБайт	rw- r-- r--	floresk1	psacln
_20_CM_FWPLiw9c.jpg	9/04/2020 18:36	4.0 КБайт	rw- r-- r--	floresk1	psacln
_20_1_phelegthon.jpg	9/04/2020 18:36	4.0 КБайт	rw- r-- r--	floresk1	psacln
_4_hamster.jpg	8/04/2020 18:59	4.0 КБайт	rw- r-- r--	floresk1	psacln
hamster.jpg	8/04/2020 17:38	100.0 КБайт	rw- r-- r--	floresk1	psacln
_16_bad_boy.jpg	6/04/2020 16:29	4.0 КБайт	rw- r-- r--	floresk1	psacln
avatar.jpg	5/04/2020 15:50	8.0 КБайт	rw- r-- r--	floresk1	psacln
no-avatar-png.png	4/04/2020 16:27	4.0 КБайт	rw- r-- r--	floresk1	psacln
test_1.jpg	3/03/2020 15:42	4.0 КБайт	rw- r-- r--	floresk1	psacln
test.jpg	28/02/2020 14:39	8.0 КБайт	rw- r-- r--	floresk1	psacln

Блок рисунков 3.15 – Содержимое папки «images_emp» и пути к фотографиям сотрудников

Помимо этого, все прикрепленные файлы к задачам хранятся на хостинге в папке «docx_dir», а пути к файлам задач хранятся в базе данных в поле «name_file» таблицы «tasks» (рисунок 3.16).

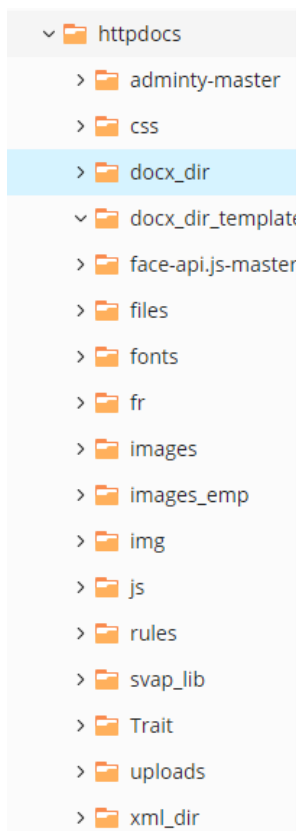


Рисунок 3.16 – Расположение папки «docx_dir»

Содержимое папки «docx_dir» и пути к прикрепленным файлам задач сотрудников, хранящиеся в самой базе данных в поле «name_file» таблицы «tasks», приведены в блоке рисунков 3.17.

Домашняя папка > httpdocs > docx_dir >

Имя	Дата последнего изменения	Размер	Права	Пользователь	Группа
..	28/04/2020 17:14		rw- r-x ---	floresk1	psaserv
Документ Microsoft Word.docx	20/05/2020 05:02	16.0 КБайт	rw- r-- r--	floresk1	psacln
logo.png	8/04/2020 17:40	0 Байт	rw- r-- r--	floresk1	psacln
111.mxl	8/04/2020 17:09	0 Байт	rw- r-- r--	floresk1	psacln
task-list-2.php	7/04/2020 16:17	52.0 КБайт	rw- r-- r--	floresk1	psacln
AMMY_service.log	7/04/2020 15:48	0 Байт	rw- r-- r--	floresk1	psacln
AA_v3.log	7/04/2020 15:43	0 Байт	rw- r-- r--	floresk1	psacln
11_20200331-02D33.txt	5/04/2020 17:54	0 Байт	rw- r-- r--	floresk1	psacln
1234.txt	23/03/2020 18:18	4.0 КБайт	rw- r-- r--	floresk1	psacln

	id_tasks_1	name_tasks	id_sender	id_receiver	date_create	dead_line	description	name_file
Изменить Копировать Удалить	56	TEST	19	14	2020-04-28 22:54:49	2020-05-02	Выполнить отчет за месяц	Нет файла
Изменить Копировать Удалить	57	Выполнить отчет	14	19	2020-04-28 22:56:33	2020-04-30	Закончить отчет	Нет файла
Изменить Копировать Удалить	58	test	4	20	2020-05-20 11:02:30	2020-05-22	отчет по продаже	../httpdocs/docx_dir/Документ Microsoft Word.docx

Блок рисунков 3.17 – Содержимое папки «docx_dir» и пути к прикрепленным файлам задач сотрудников

Шаблоны, использующиеся при генерации документов (на принятие на работу, увольнение, отпуск и премию), хранятся на хостинге в папке «docx_dir_templates» (рисунок 3.18).

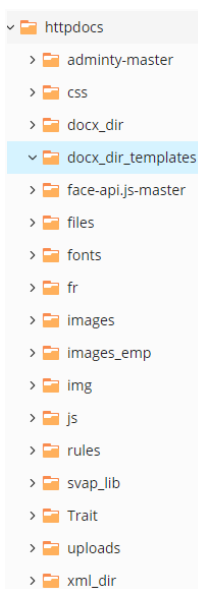
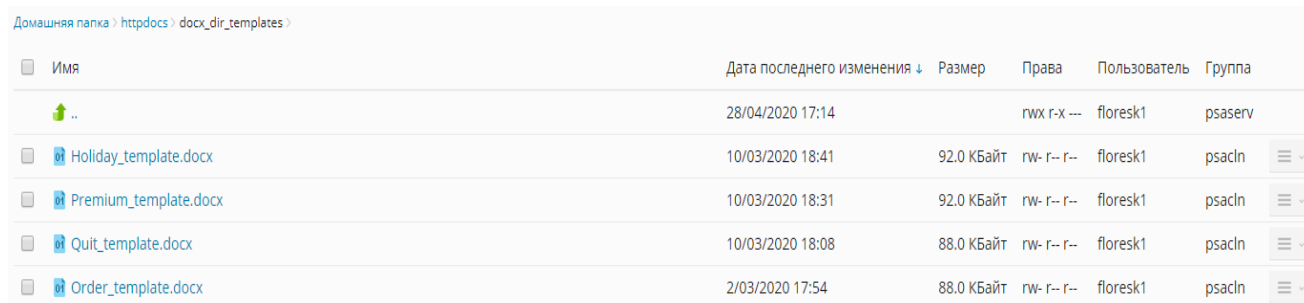


Рисунок 3.18 - Расположение папки «docx_dir_templates»

Содержимое папки «docx_dir_templates», расположенной на хостинге, представлено на рисунке 3.19.



Имя	Дата последнего изменения ↓	Размер	Права	Пользователь	Группа
..	28/04/2020 17:14		rw-x---	floresk1	psaserv
Holiday_template.docx	10/03/2020 18:41	92.0 КБайт	rw-r--r--	floresk1	psacln
Premium_template.docx	10/03/2020 18:31	92.0 КБайт	rw-r--r--	floresk1	psacln
Quit_template.docx	10/03/2020 18:08	88.0 КБайт	rw-r--r--	floresk1	psacln
Order_template.docx	2/03/2020 17:54	88.0 КБайт	rw-r--r--	floresk1	psacln

Рисунок 3.19 – Содержимое папки «docx_dir_templates»

Также, phrmyadmin содержит в себе инструменты для экспорта и импорта всей базы данных, либо же, для одной или нескольких таблиц. Данная функция очень удобна для переноса таблиц или данных из одной базы данных в другую, или же, при заполнении таблиц базы данных, допустим, из файла с расширением .txt или .xlsx. Скриншоты данных утилит приведены в блоке рисунков 3.20.

Импорт в базу данных "floresk1_diplom"

Импортируемый файл:

Файл может быть сжат в архив (gzip, zip) или находиться без сжатия.

Имя скаемого файла должно заканчиваться в виде `.[формат].[сжатие]`. Пример: `.sql.zip`

Обзор вашего компьютера: Файл не выбран (Максимальный размер: 2.048МБ)

Вы также можете просто перетащить файл на любой странице.

Кодировка файла:

Частичный импорт:

Разрешить скрипту разбивать процесс импорта при приближении временного лимита. (Может быть использовано при импорте файлов большого размера, однако при этом вероятны проблемы с транзакциями.)

Пропустить указанное число запросов (для SQL), начиная со следующего:

Прочие параметры:

Включить проверку внешних ключей

Формат:

Параметры формата:

Режим совместимости SQL:

Не использовать атрибут `auto_increment` для нулевых значений

Экспорт таблиц из базы данных "floresk1_diplom"

Экспорт шаблонов:

Новый шаблон:

Имеющиеся шаблоны: Шаблон:

Метод экспорта:

Быстрый - отображать минимум настроек

Обычный - отображать все возможные настройки

Формат:

Блок рисунков 3.20 – Импорт и экспорт базы данных

Импорт в базу данных можно выполнить в доступных форматах, изображенных на рисунке 3.21.

Формат:

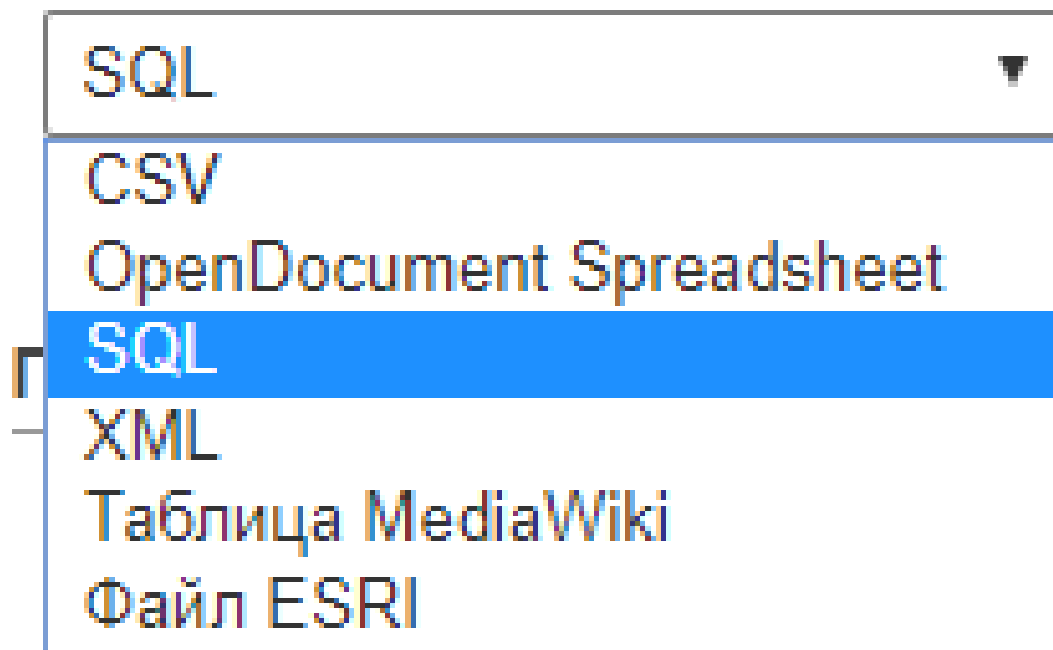


Рисунок 3.21 – Доступные форматы для импорта данных в базу данных

Экспорт из базы данных можно выполнить в доступных форматах, изображенных на рисунке 3.22.

Формат:

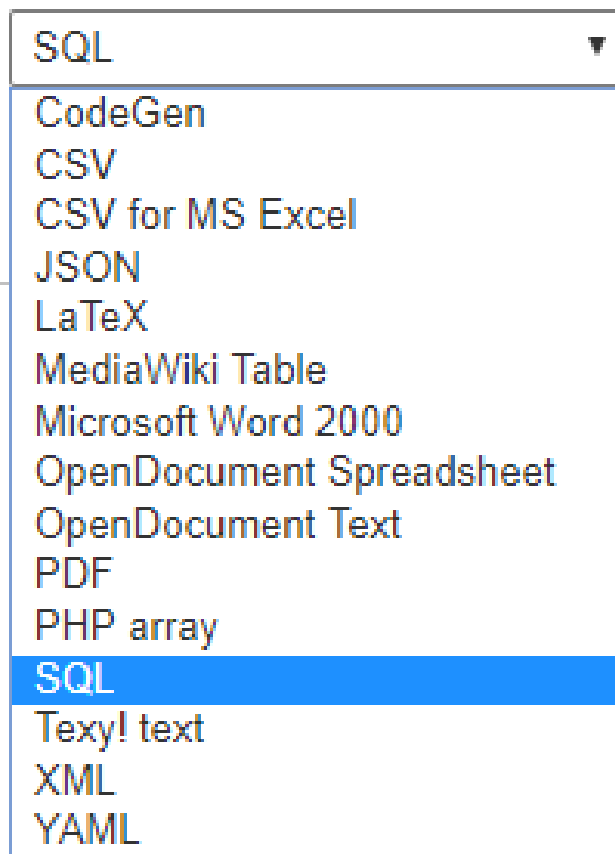


Рисунок 3.21 – Доступные форматы для экспорта данных из базы данных

4 Безопасность жизнедеятельности

4.1 Анализ потенциально опасных и вредных факторов в офисе, воздействующих на персонал

Тема данного дипломного проекта – разработка системы хранения и обработки цифровых данных. Работа любого программиста напрямую связана с использованием компьютера, а соответственно с опасным воздействием группы факторов, что значительно снижает производительность их труда. Компьютерная техника наносит огромный урон здоровью человека, поэтому работодателю необходимо учесть все вредные факторы на рабочем месте и/или при работе с компьютерной техникой для комфортной и безопасной работы своего сотрудника.

Начнем с описания рабочего места:

- рабочее помещение располагается на 3 этаже;
- тип аудитории: офисное помещение;
- помещение представляет собой комнату, размеры которой 6×8×3.2 (ширина-длина-высота);
- источники света: тип светильника ЛЛЦК-40;
- два окна (2x1,75 м каждое), оба окна расположены на южной стороне;
- кабинет и мебель окрашены по стандарту, что способствует созданию благоприятных условий для зрительного восприятия;
- для защиты от избыточной яркости с окон установлены жалюзи;
- в помещении находятся 4 человека, которые работают в дневную смену 5 дней в неделю (режим работы с 9:00 до 18:00).

Оборудование, применяемое в помещении:

- персональные компьютеры в количестве 4 штук типа IBM PC со следующими характеристиками:

HP Gen10 (umTower 4LFF)/2-Core AMD Opteron X3216(1.6GHz)/8-32GBUDIMM/4xLFF+1xSSD(nhp)/RAID0,1,10/2x1GbE/200W/19"LG1950/Клавиатура/ Мышь;

- 1 маршрутизатор типа Cisco 7300;
- 1 коммутатор типа Cisco 4700;
- в качестве осветительных приборов используются люминесцентные лампы, мощностью 40 Вт каждая.

Существуют несколько широко известных вредных факторов, воздействующих на работников в процессе работы с компьютерной техникой, такой как ВДТ- видео дисплейные терминалы и персональные компьютеры:

- слабая освещенность помещения;
- вредное воздействие электромагнитных волн статического электричества;
- психоэмоциональное напряжение;
- несоответствующий микроклимат помещений.

Одним из наиболее важных факторов, влияющих на работоспособность и самочувствие сотрудника является освещенность. Нормы освещения в офисных помещениях представлены в СН РК 2.04-XX-2011. Рекомендованные значения можно найти в таблице 1 указанного документа. В ней приводятся следующие показатели для офисов:

- естественное освещение (КЕО, e_n) составляет 4,0 при верхнем или комбинированном освещении и 1,5 при боковом.

- показатель КЕО при совмещенном освещении равен 2,4 при комбинированном или верхнем свете и 0,9 при боковом.

- искусственная освещенность равна 200 лк при комбинированном и 300 лк при общем освещении.

- показатель дискомфорта М не может превышать уровня 24.

- допустимый коэффициент пульсации света составляет не более 20 %.

Микроклимат также может пагубно влиять на состояние и работу организма. Например, высокие температуры вызывают быструю усталость, перегрев организма, а также повышенное потоотделение. Низкие температуры способствуют общему или местному переохлаждению организма, и также могут стать причиной различных хронических заболеваний. На основе документа СНиП РК 4.02-42-2006 для данной категории работ (легкая Ia) применяются следующие оптимальные нормы:

- температура в рабочем помещении: 23 – 25 $^{\circ}\text{C}$;

- скорость движения воздуха в рабочем помещении, не более: 0,1 м/с;

- относительная влажность в рабочем помещении составляет: 40 – 60 %.

В любом офисном помещении, независимо от его размеров, обязательно должны быть средства пожаротушения. Например, в большом офисе необходимы собственные планы эвакуации, огнетушители, пожарная сигнализация. Основные требования, предъявляемые практически ко всем офисным помещениям, соблюдены, а именно:

- присутствие инструкций о мерах пожарной безопасности и плана эвакуации при пожаре;

- присутствие систем и средств пожаротушения (огнетушители, пожарные краны ПК и пожарные щиты) в соответствии с классификацией помещения и его особенностями;

- наличие пожарной сигнализации;

- наличие легкодоступных путей для эвакуации в случае пожара.

Соблюдение электробезопасности очень важно для сотрудников, иначе несоблюдение самых элементарных правил электрической безопасности может стать основной причиной травматизма на рабочем месте. Соблюдены следующие требования:

- отсутствие шнуров питания, изоляция которых имеет небольшие трещины и другие повреждения;

- все электрошнуры не должны быть очень короткими и слишком длинными, в особенности категорически запрещается прибивать шнуры питания к полу гвоздями, а также накрывать их ковром, линолеумом и другими напольными покрытиями;

- все розетки в офисном помещении, независимо от того, где они установлены, нельзя перегружать, т.к. они рассчитаны на определённую величину нагрузки, и могут стать причиной пожара в случае несоблюдения правил.

Соблюдение вышеуказанных правил и норм позволит сделать рабочее место максимально комфортным и безопасным для сотрудника (план офисного помещения приведен на рисунке 4.1). В частности, при организации рабочего места программиста соблюдены следующие основные условия:

- оптимальное размещение оборудования, входящего в состав рабочего места (4 ПК, 1 коммутатор, 1 кондиционер и 1 маршрутизатор);

- достаточное рабочее пространство, позволяющее осуществлять все необходимые движения и перемещения на рабочем месте;

- необходимое естественное и искусственное освещение для успешного выполнения поставленных задач;

- условия микроклимата не вызывают дискомфорта при работе;

- соблюдены необходимые условия по пожарной безопасности и электробезопасности.

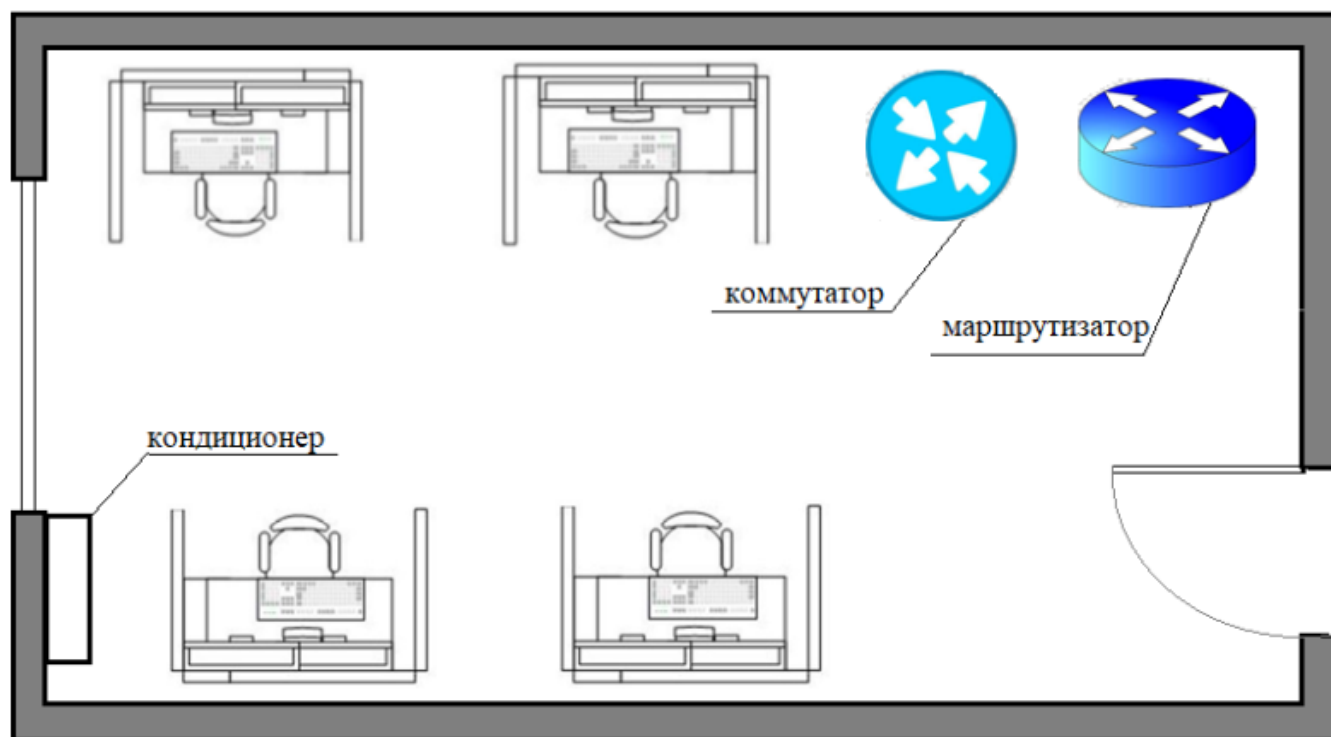


Рисунок 4.1 – План офисного помещения

4.2 Расчет естественного и искусственного освещения помещения

Расчет естественного освещения помещения

Исходные данные:

- параметры помещения (L x B x H), м: 6 x 8 x 3,2;
- высота окна $h_{ок}$, м: 2,2;
- высота начала окна $h_{н.ок.}$, м: 1,0;
- уровень условной рабочей поверхности $h_{пов.}$, м: 0,9;
- разряд зрительной работы: IV, А;
- световой пояс: г. Алматы, IV;
- $H_{зд}$, м: 24;
- расстояние до рядом стоящего здания, Р, м: 20;
- $\rho_{ср}$: 0,5;

Вышеуказанные константы взяты из СНиП РК 4.02-42-2006 [с. 17 таблица 4 и с. 6 таблица 1]. Расчет естественного освещения заключается в определении площади световых проемов.

Общую площадь окон определяем по формуле 4.1 для бокового освещения:

$$S_0 = \frac{S_n \cdot e_n \cdot \eta_0 \cdot K_{зд} \cdot K_3}{100 \cdot \tau_0 \cdot \tau_1} \quad (4.1)$$

Где: S_n – площадь пола помещения, $S_n=6 \cdot 8=48 \text{ м}^2$;
 e_n – нормированное значение КЕО (формула 4.2):

$$e_n^{IV} = e_n \cdot m \cdot c \quad (4.2)$$

Значения коэффициентов m , c находятся из таблицы 1. Значение коэффициента $e_n=1,2$ [10, с. 6 таблица 1] для работ средней точности IV подразряда. Исходя из этого, $m=0,9$; $c=0,75$.

$$e_n^{IV} = 1,2 \cdot 0,9 \cdot 0,75 = 0,81$$

K_3 – коэффициент запаса: $K_3 = 1,2$ [10, с. 13 таблица 3];

τ_0 - общий коэффициент светопропускания $\tau_0 = \tau_1 \cdot \tau_2 \cdot \tau_3 \cdot \tau_4$;

τ_1 - коэффициент светопропускания материала: для стеклопакета $\tau_1 = 0,8$ [11, таб. В48 приложения];

τ_2 - коэффициент, учитывающий потери света в переплетах светопроёма: для металлических одинарных $\tau_2 = 0,75$ [11, таб. В48 приложения];

τ_3 - коэффициент, учитывающий потери света в несущих конструкциях, при боковом освещении: для железобетонных форм $\tau_3 = 0,8$ [11, таб. В48 приложения];

τ_4 - коэффициент, учитывающий потери света в солнцезащитных устройствах: для убираться регулируемых жалюзи $\tau_4 = 1$ [11, таб. В48 приложения];

Теперь найдем τ_0 по вышеуказанным константам:

$$\tau_0 = 0,8 \cdot 0,75 \cdot 0,8 \cdot 1 = 0,48$$

η_0 - световая характеристика окон:

Отношение длины помещения к его глубине:

$$\frac{8}{6-1} = 1,3;$$

$$l=6-1=5$$

$$h_1 = h_{\text{ок}} + h_{\text{нок}} - h_{\text{пов}} = 2,2 + 1 - 0,9 = 2,3 \text{ м} \quad (4.3)$$

где h_1 – высота от уровня условной рабочей поверхности до верха окна (формула 4.3).

$$\frac{B}{h_1} = \frac{6}{2,3} = 2,6$$

Световая характеристика окон $\eta_0 = 13$ [11, таб. В46 приложения];

r_1 – коэффициент, учитывающий повышение КЕО при боковом освещении благодаря свету, отраженному от поверхностей помещения и подстилающего слоя, прилегающего к зданию:

$$\frac{B}{h_1} = \frac{6}{2,3} = 2,6$$

$$\frac{l}{B} = \frac{3}{4} = 0,75$$

$$\frac{L}{B} = \frac{6}{4} = 1,5$$

$\rho_{\text{ср}} = 0,5$ [11, таб. В45 приложения];

$r_1 = 1,5$ [11, таб. В47 приложения];

$K_{\text{зд}}$ – коэффициент, учитывающий затенение окон противостоящими зданиями (формула 4.4):

$$\frac{P}{H_{\text{зд}}} = \frac{20}{24} = 0,8 \quad (4.4)$$

$K_{\text{зд}} = 1,5$ [11, таб. В47 приложения].

Теперь подставим все значения в расчетную формулу и вычислим необходимую общую площадь окон при одностороннем боковом освещении:

$$S_0 = \frac{48 \cdot 0,81 \cdot 13 \cdot 1,5 \cdot 1,2}{100 \cdot 0,48 \cdot 0,8} \approx 23,7 \text{ м}^2$$

Отсюда следует, что для соответствия нормативам естественного освещения общая площадь окон должна быть равна $23,7 \text{ м}^2$. В нашем случае она равна 7 м^2 , таким образом, текущей площади окон недостаточно для обеспечения нормированного освещения. Поэтому выполним расчет искусственного освещения.

Расчет искусственного освещения помещения

Нормированное освещение в помещении, предназначенном для работы с ЭВМ создается при наличии как естественного, так и искусственного освещения.

Слабое освещение приводит к сильному напряжению глаз, быстрой утомляемости, близорукости, а также снижению качества работы. Яркое освещение, наоборот, раздражает сетчатку глаза и ослепляет, из-за этого глаза быстро устают и растёт производственный травматизм. Искусственное освещение нормируется посредством использования люминесцентных ламп.

В офисном помещении используются люминесцентные светильники, так как:

- имеют большую световую отдачу;
- не чувствительны к броскам тока;
- отсутствует ослепляющий эффект;
- считаются недорогими;
- имеют низкую температуру колбы.

Для создания нормальных условий, на рабочем месте проведено нормирование освещенности в зависимости от контраста объекта с фоном. Определение нормированной освещенности ведется по разрядам и подразрядам

выполняемых работ. Для работ, выполняемых программистом, отводится четвертый разряд и подразряд “А”. Минимальное значение нормированной освещенности для этих работ составляет $E_{\min} = 300$ Лк для общей системы освещения. [10, 6 с. таблица 1].

В проекте рассчитано общее освещение офисного помещения размером 6х8 м. Нормируемая освещенность составляет 300 лк.

В помещении установлена система общего освещения люминесцентными светильниками мощностью 40 Вт и световым потоком $\Phi_{\text{св}} = 4300$ Лм. Высота рабочей поверхности составляет $h_{\text{раб}} = 0.9$ м.

Индекс помещения, определяется соотношением размеров освещаемого помещения и находится по следующей формуле 4.5:

$$i = \frac{A \cdot B}{h \cdot (A + B)} \quad (4.5)$$

Далее, необходимо определить необходимое расстояние между светильниками по формуле 4.6:

$$L = \lambda \cdot h \quad (4.6)$$

где $\lambda = 1,2 \div 1,4$.

Сначала находим высоту подвеса светильников над рабочей поверхностью по формуле 4.7:

$$\begin{aligned} h &= H - h_{\text{раб}} \\ h &= 3,2 - 0,9 = 2,3 \text{ м} \end{aligned} \quad (4.7)$$

По этим данным найдем необходимое расстояние между светильниками:

$$L = 1,3 \cdot 2,3 = 2,99 \text{ м} \approx 3 \text{ м}$$

Теперь можем найти индекс помещения:

$$i = \frac{8 \cdot 6}{2,3 \cdot (8 + 6)} = 1,4907$$

В качестве светильника был взят ЛЛЦК-40 мощностью 40 Вт, световой поток которого составляет 4200 Лм, $\Phi_{\text{св}} = 4200$ лм.

Теперь найдем необходимое число светильников по формуле 4.8:

$$N = \frac{E \cdot K_3 \cdot S \cdot Z}{n \cdot \Phi_{\text{л}} \cdot \eta} \quad (4.8)$$

где S – площадь помещения, $S=8 \cdot 6=48 \text{ м}^2$;

K_3 – коэффициент запаса, $K_3 = 1,5$ [10, с.17];

E – заданная минимальная освещенность, $E = 300 \text{ лк}$ [10, 6 с. таблица 1];

Z – коэффициент неравномерности освещения (для люминесцентных ламп $Z = 1,1$);

n – количество светильников, $n = 1$;

$\Phi_{\text{л}}$ – световой поток выбранной лампы, $\Phi_{\text{св}} = 4200 \text{ лм}$;

η – коэффициент использования, $\eta = 0,61$ [11, таб. Б57 приложение].

Подставляем значения и выполняем расчет:

$$N = \frac{300 \cdot 1,5 \cdot 48 \cdot 1,1}{1 \cdot 4200 \cdot 0,61} = 9,06 \cong 9 \text{ шт.}$$

Размещаем 3 ряда светильников по 3 светильника в ряду с расстоянием между светильниками в ряду 2 метра. Всего для создания нормируемой освещенности 300 лк необходимо 9 светильников типа ЛЛЦК-40, как показано на рисунке 4.2.

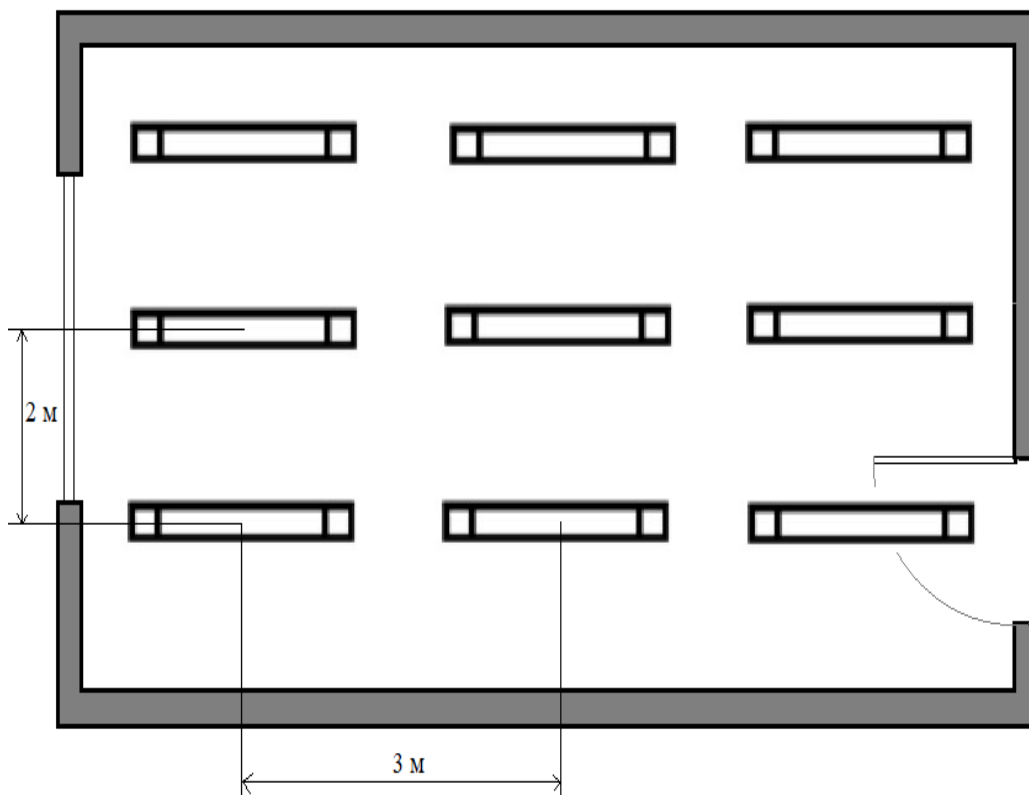


Рисунок 4.2 – Размещение светильников в офисном помещении

4.3 Расчет системы кондиционирования офисного помещения

В данном подразделе приведены расчеты системы кондиционирования в офисном помещении. Кондиционирование обеспечивает наилучшее состояние микроклимата в рабочем помещении и условия работы точной и чувствительной аппаратуры.

Для начала нужно определить количество воздуха L м³/ч, которое необходимо вывести за один час из помещения, чтобы вместе с ним удалить избыток тепла $Q_{изб}$. Количество воздуха определяется по следующей формуле 4.9:

$$L = \frac{Q_{изб}}{C_B \cdot t \cdot \gamma_B} \quad (4.9)$$

где C_B – теплоемкость сухого воздуха, ккал/кг, $C_B = 0,24$ ккал/кг [12, с. 15];

$t = t_{yx} - t_{vx}$ при расчетах возьмем $t = 10^\circ$ С [12, с. 48];

γ_B – плотность уходящего воздуха, определяемая в зависимости от температуры, кг/м³, при расчетах принимается $\gamma_B = 1.20$ кг/м³ [12, с. 40, таблица П2.1].

Далее, необходимо определить избыточное тепло $Q_{изб}$ ккал/ч (формула 4.10):

$$Q_{изб} = Q_{п} - Q_{от} \quad (4.10)$$

где $Q_{п}$ – количества тепла поступающего в воздух помещения, ккал/ч;

$Q_{от}$ – теплоотдача в окружающую среду через наружные ограждения (в теплое время года, при расчетах можно принять нулю).

Количество тепловыделений $Q_{п}$ зависит от мощности оборудования, числа работающих людей и тепла, которое вносится в помещение через оконные проемы (формула 4.11):

$$Q_{п} = Q_{об} + Q_{л} + Q_{р} + Q_{осв} \quad (4.11)$$

где $Q_{об}$ – тепло, выделяемое производственным оборудованием, ккал/ч;

$Q_{л}$ – тепло, выделяемое людьми, ккал/ч;

$Q_{р}$ – тепло, вносимое солнечной радиацией, ккал/ч.

Тепло, выделяемое производственным оборудованием в рабочем помещении, определяется по формуле 4.12:

$$Q_{об} = 860 \cdot P_{об} \cdot n \cdot N \quad (4.12)$$

где 860 – тепловой эквивалент 1 кВт·ч, то есть тепло, эквивалентное 1 кВт·ч электрической энергии;

$P_{об}$ – мощность, потребляемая оборудованием 450. $P_{об} = 450$ Вт;

n – коэффициент перехода тепла в помещение, $n = 0,75$ [13, с. 25, таблица 1];

N – количество производственного оборудования, $N=6$.

Подставим данные в формулу и найдем $Q_{об}$:

$$Q_{об} = 860 \cdot 450 \cdot 0,75 \cdot 6 = 1741,5 \text{ ккал/ч}$$

Тепло, вносимое солнечной, радиацией, определяется из соотношения (формула 4.13):

$$Q_p = m \cdot F \cdot g_{ост} \quad (4.13)$$

где m – количество окон в помещении;

F – площадь одного окна, $F = 2 \cdot 1,75 = 3,5 \text{ м}^2$;

$g_{ост}$ – солнечная радиация через остекленную поверхность, то есть количество тепла, вносимое за 1 ч через остекление площадью в 1 м^2 [13, 20 с.].

Подставляем константы и рассчитываем:

$$Q_p = 2 \cdot 3,5 \cdot 145 = 1015 \text{ ккал/ч}$$

Количество тепла, выделяемого находящимися в помещении людьми, находится по формуле 4.14:

$$Q_l = N \cdot q_l \quad (4.14)$$

где N – количество людей в помещении, $N=4$;

q_l , ккал/ч – тепловыделения одного человека, $q_l = 60$ [14, с. 49, таблица П10.1].

Рассчитаем:

$$Q_l = 4 \cdot 60 = 240 \text{ ккал/ч}$$

Количество тепла, выделяемого системой освещения, находится по формуле 4.15:

$$Q_{осв} = 860 \cdot P_{осв} \cdot \alpha \cdot b \cdot \cos(\varphi) \quad (4.15)$$

где α – коэф. перевода электрической энергии в тепловую, люминесцентные

светильники $\alpha = 0.46 \div 0.48$ [14, с. 53, таблица П14.1];
 b – коэффициент одновременности работы (при работе всех светильников $b = 9$);

$\cos(\varphi) = 0.7 \div 0.8$ – коэффициент мощности [14, с. 27 таблица 7];

$P_{осв}$, кВт – мощность осветительной установки, $P_{осв} = 40$ Вт.

Подставим вышеуказанные значения и рассчитаем:

$$Q_{осв} = 860 \cdot 0,04 \cdot 0,48 \cdot 9 \cdot 0,8 = 118,9 \text{ ккал/ч}$$

Теперь можно рассчитать тепловыделение:

$$Q_{изб} = 1741,5 + 1015 + 240 + 118,9 = 3115,4 \text{ ккал/ч}$$

Затем, необходимо рассчитать необходимый воздухообмен:

$$L = \frac{3115,4}{0,24 \cdot 10 \cdot 1,2} = 1081,7 \text{ м}^3/\text{ч}$$

Отношение количества воздуха, поступающего в помещение за один час к объему помещения, называется кратностью воздухообмена и определяется по формуле 4.16:

$$K = \frac{L}{V_{п}} \quad (4.16)$$

где $V_{п}$ – объем помещения, $V_{п} = 6 \times 8 \times 3.2 = 154 \text{ м}^3$.

Рассчитаем:

$$K = \frac{1081,7}{154} = 7$$

Теперь можно найти требуемую производительность кондиционера по нижеуказанной формуле 4.17:

$$W_k = k_3 \cdot L \quad (4.17)$$

где k_3 – коэффициент запаса, $k_3 = 1,3 \div 2,0$ [14, с.17].

Подставляем значения и рассчитаем производительность кондиционера:

$$W_k = 2 \cdot 1081,7 = 2163,4 \text{ м}^3/\text{ч}$$

Исходя из вышеуказанных расчетов в помещение с оборудованием, для соблюдения требуемых параметров микроклимата следует установить один кондиционер с производительностью не менее 2163,4 м³/ч. На основе этих данных выбран кондиционер LG P18SP Dual INV, расход воздуха (внутр./внешн.), м³/ч: 850/3110. Фото кондиционера представлено ниже (рисунок 4.3):



Рисунок 4.3 – Кондиционер LG P18SP Dual INV

4.4 Вывод по разделу безопасность жизнедеятельности

В результате проделанных расчетов рассчитаны все необходимые меры безопасности и условия труда программиста для его комфортной и безопасной работы. В частности, был произведен расчет необходимой естественной и искусственной освещенностей рабочей поверхности. И на основе этих данных, был выбран тип источника света с учетом нормированной освещенности. И затем, рассчитано и показано расположение световых источников. Искусственное и естественное освещение находятся в пределах нормы для безопасной и комфортной работы за персональным компьютером. Также было произведено нормирование показателей микроклимата, и исходя из этих расчетов, было выяснено что в офисное помещение с оборудованием, для соблюдения требуемых параметров микроклимата, требуется установить один кондиционер с производительностью не менее 2163,4 м³/ч.

5 Технико-экономическое обоснование проекта

5.1 Цель проекта

Цифровая информация в современных реалиях превратилась в один из наиболее ценных ресурсов, а базы данных (БД) стали необходимым инструментом практически в любых сферах деятельности. Каждая база данных представляет собой упорядоченную структуру, предназначенную для хранения информации и данных. Таким образом, БД играют огромную роль, поскольку эффективность работы любых предприятий, компаний или же учреждений зависит во многом от них. Любая база данных должна выполнять следующие функции:

- обеспечивать создание общих и/или детализированных отчетов по итогам проделанной работы;
- обеспечивать сбор и хранение информации без существенных задержек;
- обеспечение относительно легкого определения тенденций изменений важнейших показателей;
- выполнение точного и подробного анализа хранимых данных.

Целью данного дипломного проекта является разработка системы хранения и обработки цифровых данных для флористической компании, т.е. базы данных (БД). Эта база данных была создана в СУБД MySQL на арендуемом казахстанском хостинге, для удаленного взаимодействия и управления использовалось бесплатное веб-приложение phpmyadmin. В этой базе данных были созданы следующие таблицы: для хранения текущих данных о всех сотрудниках компании и их должностях, данных для каждого из трех типов аутентификации на информационном портале, информации, используемой при оформлении документов на увольнение, принятие на работу, отпуск и премиальные, хранения текстовых сообщений по внутреннему мессенджеру между сотрудниками компании, всех поставленных и выполненных задач, включая их статусы, хранения текущего курса иностранных валют по отношению к тенге, а также данных, получаемых с сервера 1С.

5.2 Реализация проекта

Для успешной реализации данного проекта необходимо:

- время;
- оборудование;
- человеческие ресурсы.

План разработки и реализации проекта состоит из 5 этапов, выполняемых в течение 2 месяцев (8 недель). Для удобства, план разработки и реализации проекта был разделен на 5 этапов. В первом этапе происходит аналитическая разработка

необходимых требований и условий к разрабатываемому продукту. Первый этап состоит из:

- а) постановка задачи;
- б) разработка условий к скорости работы БД, масштабируемости и др.;

На втором этапе необходимо определить все необходимые данные для разработки БД, а именно: типы данных, структуру данных, таблицы и т.д. Второй этап включает в себя:

- а) определение характеристик хранимой информации;
- б) определение структуры БД;
- в) определение таблиц;
- г) определение типов данных;

Третий этап начинается после того как таблицы в БД создались. После этого начинается создание необходимых внешних ключей, первичных ключей и прав доступа для пользователей БД. Третий этап – это:

- а) создание индексов;
- б) создание отношений;
- в) определение прав доступа пользователей.

Затем, на четвертом этапе, происходит создание логической и физической моделей в программе AllFusion ERwin Data Modeler 7, UML диаграмм в программе IBM Rational Rose XDE (версия 7.0), а также триггеров для уже созданных таблиц. На четвертом этапе производится:

- а) создание физической и логической моделей БД;
- б) создание UML диаграммы БД;
- в) создание базы данных;
- г) создание триггеров БД.

И, наконец, на пятом этапе разработки происходит тестирование разработанного продукта на наличие ошибок, а также внедрение на предприятие. На пятом этапе происходит:

- а) тестирование;
- б) внедрение.

Для удобства рассмотрения вышеизложенных этапов разработки была составлена таблица, в которой желтым цветом обозначены потраченные недели на выполнение каждого из вышеприведенных этапов.

План разработки и реализации проекта представлен в таблице 5.1.

Таблица 5.1 - План разработки и реализации проекта

Перечень работ		Недели от начала работ							
		1	2	3	4	5	6	7	8
1 этап	постановка задачи								
	разработка условий к скорости работы БД, масштабируемости и др.								

Продолжение таблицы 5.1

2 этап	определение характеристик записываемой информации								
	определение структуры БД								
	определение таблиц								
	определение типов данных								
3 этап	создание индексов								
	создание отношений								
	определение прав доступа пользователей								
4 этап	создание физической и логической моделей БД								
	создание UML диаграммы БД								
	создание базы данных								
	создание триггеров БД								
5 этап	тестирование								
	внедрение								

Количество и заработная плата работников, задействованных в разработке, представлены в таблице 5.2.

Таблица 5.2 – Заработная плата работников

Исполнители	Количество, человек	Зар. плата за час, тенге	Зар. плата за день, тенге	Зар. плата за месяц, тенге
Инженер БД	1	800,00	6400,00	140800,00
Итого:	1	800,00	6400,00	140800,00

Длительность цикла в днях по каждому из всех видов работ рассчитывается по следующей формуле 5.1:

$$t_n = \frac{T}{q_n \cdot z \cdot K}, \quad (5.1)$$

где T – трудоёмкость этапа (норма-час);

q_n – количество исполнителей по этапу;

z – продолжительность рабочего дня, $z = 8$ часов;

K – коэффициент выполнения норм времени, $K = 1,1$ [15].

Полученную величину t_n нужно округлить в большую сторону до целых дней.

5.3 Расчёт стоимости произведенного проекта

Разработка данной базы данных – сложный и трудоёмкий процесс, требующий интеллектуальные, технические и финансовые затраты. Поэтому необходимо составить смету затрат на разработку базы данных. Затраты на разработку определяются по формуле 5.2:

$$C = \text{ФОТ} + \sum H + A + H \quad (5.2)$$

Где: ФОТ – фонд оплаты труда (основная и дополнительная заработная плата);

$\sum H_a$ – сумма налогов;

A – амортизационные отчисления;

H – накладные расходы (75% от всех затрат) [15].

5.4 Расчёт затрат на оплату труда

Затраты на оплату труда персонала, задействованного в разработке программного продукта (приведены в таблице 5.3), рассчитываются по формуле 5.3:

$$\text{ФОТ} = Z_{\text{осн}} + Z_{\text{доп}} \quad (5.3)$$

где: $Z_{\text{осн}}$ – основная заработная плата;

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

Средний дневной заработок работника за один рабочий день рассчитывается по формуле 5.4:

$$D = \frac{Z_{\text{Пм}}}{D_{\text{р}}} \quad (5.4)$$

Где: $Z_{\text{Пм}}$ – ежемесячный размер заработной платы;

$D_{\text{р}}$ – количество рабочих дней в месяце (это 22 дня – пятидневная рабочая неделя) [15].

Для инженера БД:

$$D = \frac{140800}{22} = 6400,00 \text{ тенге}$$

Таблица 5.3 – Расчёт основной заработной платы персонала

	Наименование этапов и содержание работ	Исполнитель	Трудоемкость		Длительность исполнения, часы	Зарплата за час работы, тенге	Сумма зарплат, тенге
			Норма-часы	% от общей трудоемкости			
1	постановка задачи	Инженер БД	4	2,27		800,00	3200,00
2	разработка условий к скорости работы БД, масштабируемости и др.	Инженер БД	2	1,14	8	800,00	1600,00
3	определение характеристик записываемой информации	Инженер БД	10	5,68		800,00	8000,00
4	определение структуры БД	Инженер БД	8	4,54		800,00	6400,00
5	определение таблиц	Инженер БД	8	4,54		800,00	6400,00
6	определение типов данных	Инженер БД	8	4,54		800,00	6400,00
7	создание индексов	Инженер БД	6	3,4	16	800,00	4800,00
8	создание отношений	Инженер БД	8	4,54	24	800,00	6400,00
9	определение прав доступа пользователей	Инженер БД	8	4,54		800,00	6400,00
10	создание физической и логической моделей БД	Инженер БД	10	5,68	24	800,00	8000,00

Продолжение таблицы 5.3

11	создание UML диаграммы БД	Инженер БД	8	4,54	24	800,00	6400,00
12	создание базы данных	Инженер БД	30	17,1	200	800,00	24000,00
13	создание триггеров БД	Инженер БД	20	11,36	32	800,00	16000,00
14	тестирование	Инженер БД	6	3,4	32	800,00	4800,00
15	внедрение	Инженер БД	40	22,73		800,00	32000,00
	Итого:		176	100,00	360		140800,00

Фонд оплаты труда (ФОТ) состоит из основной и дополнительной заработной платы:

$$\text{ФОТ} = Z_{\text{осн}} + Z_{\text{доп}}$$

Основная заработная плата определяется как сумма оплаты труда на всех стадиях разработки проекта (таблица 5.3).

$$Z_{\text{осн}} = 140800,00 \text{ тенге}$$

Дополнительная заработная плата (премии и т. д.) составляет 10% от основной заработной платы по формуле 5.5:

$$Z_{\text{доп}} = Z_{\text{осн}} \cdot 10\% \quad (5.5)$$

$$Z_{\text{доп}} = 140800,00 \cdot 10\% = 14080 \text{ тенге}$$

$$\text{ФОТ} = 140800,00 + 14080,00 = 154880,00 \text{ тенге}$$

Налоги для юридического лица рассчитываются по следующей формуле 5.6:

$$\sum H = (CH + CO + BOCMC) \quad (5.6)$$

Где CH – социальный налог;
CO – социальные отчисления;
BOCMC - Отчисления на ОСМС.

Также необходимо рассчитать ОПВ (обязательный пенсионный взнос) по формуле 5.7:

$$\text{ОПВ} = \text{ФОТ} \cdot 10\% ; \quad (5.7)$$

$$\text{ОПВ} = 154880,00 \cdot 10\% = 15488 \text{ тенге}$$

Далее рассчитываем ВОСМС (Взнос на ОСМС работником) по формуле 5.8:

$$\text{ВОСМС} = 2\% \cdot \text{ФОТ}; \quad (5.8)$$

$$\text{ВОСМС} = 154880,00 \cdot 2\% = 3097,60 \text{ тенге}$$

Затем, рассчитаем СО по формуле 5.9:

$$\text{СО} = (\text{ФОТ} - \text{ОПВ}) \cdot 3,5\% \quad (5.9)$$

$$\text{СО} = (154880,00 - 15488,00) \cdot 3,5\% = 4878,72 \text{ тенге}$$

После этого, рассчитываем СН по формуле 5.10:

$$\text{СН} = (\text{ФОТ} - \text{ОПВ} - 1\% \cdot \text{ФОТ}) \cdot 9,5\% - \text{СО} \quad (5.10)$$

$$\text{СН} = (154880,00 - 15488,00 - 1548,80) \cdot 0,095 - 4878,72 = 8216,38 \text{ тенге}$$

Суммируем вышеуказанные налоги:

$$\Sigma \text{Н} = (\text{СО} + \text{СН} + \text{ВОСМС}) = 4878,72 + 8216,38 + 3097,60 = 16192,7 \text{ тенге}$$

Амортизационные отчисления на ОС (основные средства) вычисляем по формуле 5.11:

$$A = \frac{N_{\text{АМ}} \cdot C_{\text{пер}} \cdot N}{100 \cdot 12 \cdot n} \quad (5.11)$$

где: $N_{\text{АМ}}$ – норма амортизации;

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

N - количество дней на выполнение работ.

В сумму амортизационных отчислений на ОС не входит программное обеспечение, так как сервер БД работает на операционной системе Linux, являющейся бесплатной. Веб-приложение phpmyadmin также является

бесплатным, а аренда сервера MySQL входит в стоимость хостинга (ОС приведены в таблице 5.4). Рассчитываем амортизационные отчисления на ОС:

$$A = \frac{40 \cdot 170900 \cdot 45}{100 \cdot 12 \cdot 20} = 12817,5 \text{ тенге}$$

где n=20; N =45 [16].

Таблица 5.4 – Основные средства

Наименование оборудования	Кол-во	Норма амортизации, %	Сумма амортизации, тенге	Рыночная цена за единицу, тенге	Балансовая стоимость за единицу, тенге
HP Gen10 (umTower 4LFF)/2-Core AMD Opteron X3216 (1.6GHz)/8-32GB UDIMM/4xLFF+ 1xSSD(nhp)/RAID 0,1,10/2x1GbE/200 W KeyBoard PS/2 Asus; Mouse ; Viewsonic SM713N 17''	1	40	12817,5	170900,00	196535,00
Итого:			12817,5		

Поскольку в процессе разработки базы данных используется компьютер, то необходимо рассчитать затраты на электроэнергию. Затраты на электроэнергию определим по формуле 5.12:

$$\mathcal{E} = W \cdot T \cdot S \cdot K_{\text{ИМ}} = \sum W \cdot S \quad (5.12)$$

где: W – установленная мощность приборов, потребляющих электроэнергию, кВт;

S – стоимость киловатт-часа электроэнергии (19,17 тг/кВт·ч на 2020 год);

K_{ИМ} – коэффициент использования мощности (0,9-1) [16];

T – время работы приборов, час.

Расходы на электроэнергию представлены в таблице 5.5.

Таблица 5.5 – Расходы на электроэнергию

Наименование приборов	W, кВт	Число рабочих дней	K _{им}	Время работы прибора, час	$\sum W$, кВт·ч
1 компьютер с монитором	0,2	45	0,9	360	64,8
Итого:					64,8

Затраты на электроэнергию составили:

$$\mathcal{E} = 64,8 \cdot 19,17 = 1242,216 \text{ тенге}$$

Накладные расходы на разработку программного продукта принимаем в размере 75% от общей суммы расходов и рассчитывается по нижеприведенной формуле 5.13:

$$H_p = (\text{ФОТ} + A + \mathcal{E}) \cdot 0,75 \quad (5.13)$$

$$H_p = (154880,00 + 12817,5 + 1242,216) \cdot 0,75 = 126704,78 \text{ тенге}$$

Общие затраты по разработке базы данных в соответствии с формулой составляют:

$$C = 154880,00 + 16045,57 + 12817,5 + 126704,78 + 1242,216 + 196535,00 = 508225,07 \text{ тенге}$$

Смета затрат и их структура по разработке базы данных для флористической компании по всем перечисленным статьям представлена в таблице 5.6 и на рисунке 5.1.

Таблица 5.6 – Себестоимость разработки по всем статьям затрат

Наименование статей затрат	Сумма, тенге	Структура затрат, %
ФОТ (фонд оплаты труда)	154880,00	30,47
Отчисления на социальные нужды	14643,9	3,16
Амортизация	16192,7	2,52
Затраты на электроэнергию	1242,216	0,25
Накладные расходы	126704,78	24,93
ОС (основные средства)	196535,00	38,67
Итого:	510198,6	100,00

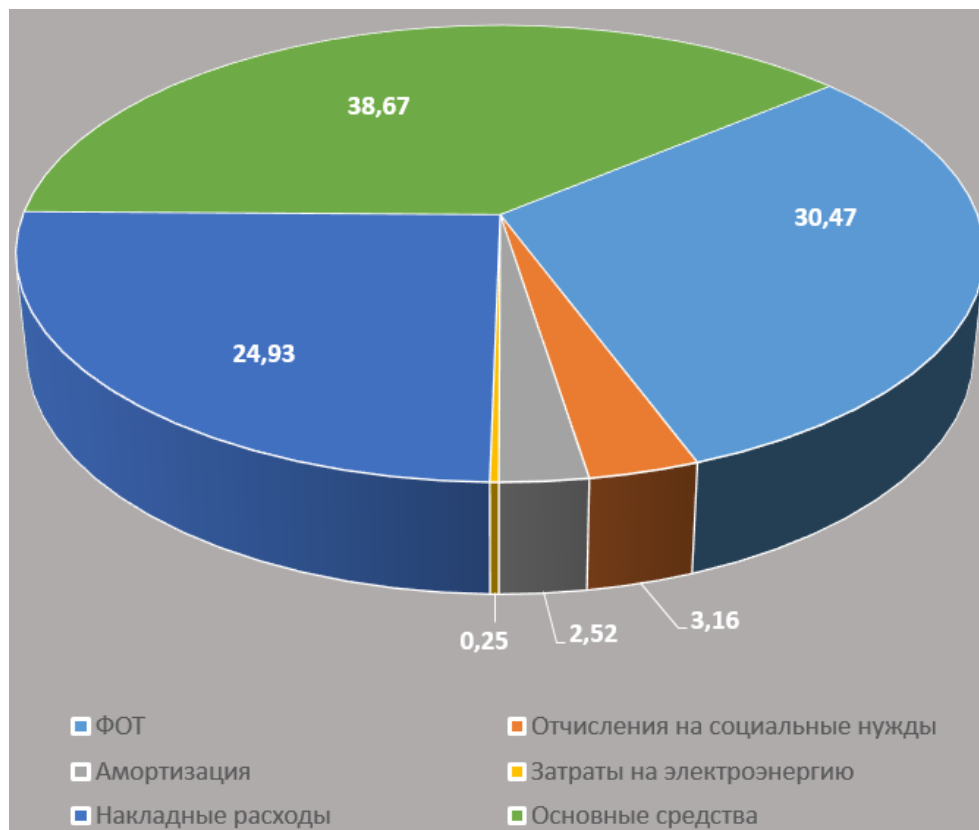


Рисунок 5.1 - Структура затрат по разработке базы данных для флористической компании

5.5 Цена реализации программного продукта

Цена реализации программного продукта складывается из себестоимости и чистого дохода по формуле 5.14:

$$Ц_{\Pi} = C + \Pi, \quad (5.14)$$

где C – себестоимость продукта;

Π – чистый доход.

При определении первоначальной цены задаёмся желаемым уровнем рентабельности ($P = 10\%$ для отрасли информационных технологий) по формуле 5.15:

$$Ц_{\Pi} = C \cdot \left(1 + \frac{P}{100}\right) \quad (5.15)$$

$$Ц_{\Pi} = 510198,6 \cdot \left(1 + \frac{10}{100}\right) = 561218,46 \text{ тенге}$$

Далее, определим цену реализации программного продукта (формула 5.16):

$$Ц_p = Ц_п + НДС \quad (5.16)$$

На сегодняшний день размер НДС в РК составляет 12% согласно НК РК [17], формула 5.17 приобретает следующий вид:

$$НДС = Ц_п \cdot 12\% \quad (5.17)$$

$$НДС = 561218,46 \cdot 0,12 = 67346,21 \text{ тенге.}$$

Теперь можно рассчитать полную себестоимость разработки базы данных:

$$Ц_p = 561218,46 + 67346,21 = 628564,67 \text{ тенге.}$$

5.6 Расчет эффективности и срока окупаемости программного продукта

До внедрения программного продукта зарплата сотрудника, занимающегося хранением и содержанием бумажных документов, составляла 180000,00 тенге/месяц. Рассчитаем годовую зарплату:

$$ФОТ = З_{\text{осн}} + З_{\text{доп}} = (180000 + (180000 \cdot 0,1)) \cdot 12 = 2376000 \text{ тенге}$$

В качестве расходуемых материалов будет использоваться бумага А4 и картридж для принтера (таблица 5.7).

Таблица 5.7 - Расходуемые материалы

Наименование	Количество, шт	Цена за 1 шт., тенге
Бумага А4, 500 листов	24	1700
Картридж для принтера	2	6000

При условии, что в месяц будет тратиться 1000 листов А4 (2 пачки бумаги), то за год будет израсходовано 24 пачки. Средний износ картриджа составляет 6500 листов, следовательно, на год хватит 2 картриджей. Рассчитаем сумму РМ за год:

$$\sum РМ = (24 \cdot 1700) + (6000 \cdot 2) = 52800 \text{ тенге}$$

В качестве ОС (основных средств) используется ноутбук и принтер (таблица 5.8).

Таблица 5.8 – Основные средства

Наименование оборудования	Кол-во	Норма амортизации, %	Сумма амортизации, тенге	Рыночная цена, тенге	Балансовая стоимость тенге
Ноутбук Asus	1	40	11625	139500,00	155000,00
Принтер Canon	1	40	4875	58500,00	65000,00
Итого:			16500 (198000 за год)		

Рассчитаем норму амортизации:

$$A = \frac{40 \cdot 155000 \cdot 45}{100 \cdot 12 \cdot 20} + \frac{40 \cdot 65000 \cdot 45}{100 \cdot 12 \cdot 20} = 11625 + 4875 = 16500 \text{ тенге}$$

Далее, рассчитаем расходы на электроэнергию за год (таблица 5.9).

Таблица 5.9 – Расходы на электроэнергию

Наименование приборов	W, кВт	Число рабочих дней	K _{им}	Время работы прибора, час	$\sum W$, кВт·ч
1 ноутбук	0,2	21	0,9	168	30,24
1 принтер	0,38	21	0,9	168	57,46
Итого:					87,7

Расходы на электроэнергию за год составили:

$$\varepsilon = \left(\sum W \cdot S \right) \cdot 12 = (87,7 \cdot 19,17) \cdot 12 = 20174,5 \text{ тенге}$$

Рассчитаем накладные расходы за год:

$$H_p = (2376000 + 198000 + 20174,5) \cdot 0,75 = 1945631 \text{ тенге}$$

После внедрения программного продукта, зарплата сотрудника снизится и составит 100000 тенге/месяц. Рассчитаем годовую ЗП:

$$\Phi OT = Z_{\text{осн}} + Z_{\text{доп}} = (100000 + (100000 \cdot 0,1)) \cdot 12 = 1320000 \text{ тенге}$$

Нужда в расходуемых материалах (бумага А4 и картриджи для принтера) и принтере отпадет, т.к. документы будут храниться в БД и все действия с

документами будут проводиться в электронном виде. Поэтому, в качестве ОС вместо ноутбука и принтера будет использоваться более мощный компьютер с сервером БД и монитором (таблица 5.10).

Таблица 5.10 – Основные средства

Наименование оборудования	Кол-во	Норма амортизации, %	Сумма амортизации, тенге	Рыночная цена, тенге	Балансовая стоимость тенге
Компьютер с монитором	1	40	18750	225000,00	250000,00
Итого:			18750 (225000 за год)		

Рассчитаем норму амортизации:

$$A = \frac{40 \cdot 250000 \cdot 45}{100 \cdot 12 \cdot 20} = 18750 \text{ тенге}$$

Далее, рассчитаем расходы на электроэнергию за год (таблица 5.11).

Таблица 5.11 – Расходы на электроэнергию

Наименование приборов	W, кВт	Число рабочих дней	K _{им}	Время работы прибора, час	$\sum W$, кВт·ч
1 компьютер с монитором	0,45	21	0,9	168	68,04
Итого:					68,04

Рассчитаем расходы на электроэнергию за год:

$$\varepsilon = \left(\sum W \cdot s \right) \cdot 12 = (68,04 \cdot 19,17) \cdot 12 = 15652 \text{ тенге}$$

Рассчитаем накладные расходы за год.

$$H_p = (1320000 + 225000 + 15652) \cdot 0,75 = 1170489 \text{ тенге}$$

Теперь посчитаем разницу годовых эксплуатационных затрат до внедрения и после внедрения программного продукта (таблица 5.12):

Таблица 5.12 - Годовые эксплуатационные затраты

Статьи	Без применения программного продукта	С применением программного продукта
Годовая заработная плата	2376000	1320000
Расходуемые материалы	52800	0
Амортизационные отчисления	198000	225000
Накладные расходы	1945631	1170489
Всего:	4572431	2715489

Далее рассчитаем годовой эффект по формуле 5.18 [17]:

$$\mathcal{E} = 4572431 - 2715489 = 1856942 \text{ тенге} \quad (5.18)$$

Теперь можно найти расчетный срок окупаемости по следующей формуле 5.19 [17]:

$$T_{\text{ок}} = C / \mathcal{E}, \quad (5.19)$$

где C - затраты на разработку и внедрение системы, тенге;
 \mathcal{E} – годовой эффект.

Подставляем значения и находим срок окупаемости:

$$T_{\text{ок}} = 628564,67 / 1856942,00 = 0,34 \text{ года.}$$

В данном случае срок окупаемости проекта составит 0,34 года или 4,08 месяца.

5.7 Вывод по технико-экономическому разделу

Разработка данной системы хранения и обработки цифровых данных – сложный и трудоёмкий процесс, требующий наряду с интеллектуальными, техническими затратами и финансовыми затратами. Затраты на разработку данной базы данных составили 510198,6 тенге.

Цена реализации программного продукта, которая состоит из его себестоимости и чистого дохода, равна 628564,67 тенге с учетом НДС, который составляет 12% согласно НК РК. Срок окупаемости проекта составит 0,34 года или 4,08 месяца.

ЗАКЛЮЧЕНИЕ

В данном дипломном проекте была рассмотрена одна из главных проблем компании – это составление и хранение бумажных документов (прием на работу, увольнение, премиальные, отпуск, данные о сотрудниках), которые приходилось создавать и заверять вручную, из-за чего и снижалась эффективность и быстродействие компании.

Во время разработки программного продукта были изучены и задействованы следующие утилиты: система управления реляционными базами данных с открытым исходным кодом MySQL (версия 10.2.31), веб-приложение с открытым исходным кодом phpmyadmin (версия 4.9.0.1), графический инструмент для моделирования логической и физической моделей баз данных AllFusion ERwin Data Modeler 7 (версия 7.0) и CASE средство проектирования и разработки UML диаграмм IBM Rational Rose XDE (версия 7.0).

Также были выполнены все поставленные задачи, а именно:

- проведен анализ предметной области;
- разработано техническое задание (ТЗ);
- выполнен сравнительный анализ существующих аналогов;
- выбрано программное обеспечение для реализации базы данных;
- спроектирована база данных;
- созданы логическая и физическая модели баз данных;
- созданы UML диаграммы классов и вариантов использования;
- выполнена программная реализация базы данных.

После внедрения разработанной системы хранения, появилась возможность сохранять все данные о сотрудниках, документы, плановые задачи, новости компании в таблицах базы данных MySQL. Экономия была достигнута путем снижения времени создания и подписания различного рода документов, которые раньше совершались вручную.

Помимо всего вышеперечисленного, были проведены расчеты необходимой естественной и искусственной освещенностей рабочей поверхности и нормирование показателей микроклимата для комфортной и безопасной работы сотрудников компании.

В технико-экономическом разделе данной дипломной работы была рассчитана стоимость разработки проекта, которая составила 510198,6 тенге, и цена реализации программного продукта, равная 628564,67 тенге с учетом НДС. Помимо всего этого, был рассчитан срок окупаемости проекта, который составил 0,34 года или 4,08 месяца, что является очень хорошим показателем.

Проводя комплексный анализ всего вышеуказанного результата, можно отметить, что цель проекта была достигнута с высокой степенью реализации, что дает основание говорить об эффективности проекта в целом.

СПИСОК ЛИТЕРАТУРЫ

- 1 Дронов, В.А. PHP, MySQL, HTML5 и CSS 3. Разработка современных динамических Web-сайтов / В.А. Дронов. - М.: БХВ-Петербург, 2016. - 226 с.
- 2 MySQL руководство администратора. - М.: Вильямс, 2010. - 621 с.
- 3 Яргер, Р.Дж. MySQL и mSQL: Базы данных для небольших предприятий и Интернета / Р.Дж. Яргер, Дж. Риз, Т. Кинг. - М.: СПб: Символ-Плюс, 2015. - 560 с.
- 4 Аткинсон, Леон MySQL. Библиотека профессионала; М.: Вильямс, 2011. - 624 с.
- 5 Архипенков, С.; Голубев, Д.; Максименко, О. Хранилища данных. От концепции до внедрения; М.: Диалог-МИФИ, 2010. – 528 с.
- 6 Грабер, Мартин SQL. Справочное руководство; М.: Лори; Издание 2-е, 2011. - 354 с.
- 7 Кузнецов Максим , Симдянов Игорь MySQL 5; БХВ-Петербург - Москва, 2010. - 502 с.
- 8 Алексей Паутов MySQL: Руководство разработчика. Версия 1.0. 20 апреля 2014 г. URL: http://citforum.ru/database/mysql/developer_man/ (дата обращения: 07.04.2020)
- 9 Marc Delisle, Mastering phpMyAdmin for effective MySQL Management; Birmingham: Packt Publishing, 2010. – 316 с. URL: <https://is.gd/scKjLm> (дата обращения: 07.03.2020)
- 10 СН РК 2.04-XX-2011 Естественное и искусственное освещение.
- 11 Ветошкин А.Г., Нормативное и техническое обеспечение безопасности жизнедеятельности. Ч. 2: учебно-практическое пособие: В 2-х ч. / Ветошкин
- 12 СНиП РК 4.02-42-2012 Отопление, вентиляция и кондиционирование.
- 13 Арустамов, Э.А. Безопасность жизнедеятельности: Учебник для бакалавров, 19-е изд., перераб. и доп.(изд:19) / Э.А. Арустамов. — М.: ИТК Дашков и К, 2016. — 448 с.
- 14 Холостова, Е.И. Безопасность жизнедеятельности: Учебник для бакалавров / Е.И. Холостова, О.Г. Прохорова. — М.: ИТК Дашков и К, 2016. — 456 с.
- 15 Аманбаев У.А. Экономика предприятия – Алматы «Бастау» 2012 г.
- 16 Бурув В. П. Бизнес план фирмы, - М., «Инфра-М» 2011г.
- 17 Куатова Д. Я. Экономика предприятия – Алматы «Экономика», 2011.

Приложение А (обязательное)
SQL код создания БД

```
CREATE TABLE `buhgalter` (  
  `id_buh` int(11) NOT NULL,  
  `name` varchar(64) NOT NULL,  
  `surname` varchar(64) NOT NULL,  
  `patronymic` varchar(64) NOT NULL,  
  `gender` varchar(11) NOT NULL,  
  `birthday` date NOT NULL,  
  `location` varchar(64) NOT NULL,  
  `date_add_buh` timestamp NOT NULL DEFAULT current_timestamp() ON  
UPDATE current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `completed_tasks` (  
  `id_task_completed` int(11) NOT NULL,  
  `id_users_tasks` int(11) NOT NULL,  
  `id_tasks_sender` int(11) NOT NULL,  
  `id_statuses` int(11) NOT NULL,  
  `date_completed` timestamp NOT NULL DEFAULT current_timestamp() ON  
UPDATE current_timestamp(),  
  `view_trigger` tinyint(1) NOT NULL DEFAULT 0  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `conversation` (  
  `c_id` int(11) NOT NULL,  
  `user_one` int(11) NOT NULL,  
  `user_two` int(11) NOT NULL,  
  `ip` varchar(30) DEFAULT NULL,  
  `time` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `conversation_reply` (  
  `cr_id` int(11) NOT NULL,  
  `reply` text NOT NULL,  
  `user_id_fk` int(11) NOT NULL,  
  `ip` varchar(30) NOT NULL,
```

```
`time` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE
current_timestamp(),
`c_id_fk` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `currency` (
`currency_id` int(11) NOT NULL,
`Name` varchar(20) NOT NULL,
`Date` date NOT NULL,
`kurs` int(11) NOT NULL,
`rate` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `director` (
`id_dir` int(11) NOT NULL,
`name` varchar(64) NOT NULL,
`surname` varchar(64) NOT NULL,
`patronymic` varchar(64) NOT NULL,
`gender` varchar(11) NOT NULL,
`birthday` date NOT NULL,
`location` varchar(64) NOT NULL,
```

Продолжение приложения А

```
`date_add_dir` timestamp NOT NULL DEFAULT current_timestamp() ON
UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `doc_job` (
`id_doc_job` int(11) NOT NULL,
`date_start` date NOT NULL,
`fio` varchar(64) NOT NULL,
`dolzh` varchar(64) NOT NULL,
`oklad` int(11) NOT NULL,
`status` tinyint(1) NOT NULL,
`date_finish` date NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `employee` (
`id_emp` int(11) NOT NULL,
`name` varchar(64) NOT NULL,
`surname` varchar(64) NOT NULL,
```

```

`patronymic` varchar(64) NOT NULL,
`gender` varchar(11) NOT NULL,
`birthday` date NOT NULL,
`location` varchar(64) NOT NULL,
`date_add_emp` timestamp NOT NULL DEFAULT current_timestamp() ON
UPDATE current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `holiday` (
`id_hol_doc` int(11) NOT NULL,
`name` varchar(64) NOT NULL,
`surname` varchar(64) NOT NULL,
`patronymic` varchar(64) NOT NULL,
`date_start_hol` date NOT NULL,
`date_end_hol` date NOT NULL,
`status_holiday` tinyint(1) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `hr_emp` (
`id_hr` int(11) NOT NULL,
`name` varchar(64) NOT NULL,
`surname` varchar(64) NOT NULL,
`patronymic` varchar(64) NOT NULL,
`gender` varchar(11) NOT NULL,
`birthday` date NOT NULL,
`location` varchar(64) NOT NULL,
`date_add_hr` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE
current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8;

```

```

CREATE TABLE `job_accepted` (
`id` int(11) NOT NULL,
`name` varchar(64) NOT NULL,
`surname` varchar(64) NOT NULL,
`patronymic` varchar(64) NOT NULL,
`date_of_birth` date NOT NULL,
`gender` varchar(16) NOT NULL,
`salary` int(11) NOT NULL,
`position` varchar(64) NOT NULL,
`iin` bigint(13) NOT NULL,

```

Продолжение приложения А

```
`date_of_start` date NOT NULL,  
`date_of_end` date NOT NULL,  
`link` varchar(128) NOT NULL,  
`status_1` tinyint(1) DEFAULT NULL,  
`cur_date` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `news` (  
  `id_news` int(11) NOT NULL,  
  `text_news` text NOT NULL,  
  `date_publish` timestamp NOT NULL DEFAULT current_timestamp() ON  
UPDATE current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `old_tasks` (  
  `id_old_tasks` int(11) NOT NULL,  
  `id_tasks_users_old` int(11) NOT NULL,  
  `id_tasks_sender_old` int(11) NOT NULL,  
  `id_statuses_old` int(11) NOT NULL,  
  `date_finish` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `old_users` (  
  `id_old_users` int(11) NOT NULL,  
  `fio_old` varchar(64) NOT NULL,  
  `u_date` datetime NOT NULL,  
  `u_id` int(11) NOT NULL,  
  `id_roles_old` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `phones` (  
  `id` int(6) UNSIGNED NOT NULL,  
  `phone` varchar(50) NOT NULL,  
  `code` varchar(48) NOT NULL,  
  `registration_date` timestamp NOT NULL DEFAULT current_timestamp() ON  
UPDATE current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `photo_emp` (  
  `id_photo` int(11) NOT NULL,  
  `id_emp` int(11) NOT NULL,  
  `picture` varchar(64) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `premium` (  
  `id_prem` int(11) NOT NULL,  
  `name` varchar(64) NOT NULL,  
  `surname` varchar(64) NOT NULL,  
  `patronymic` varchar(64) NOT NULL,  
  `prem` float NOT NULL,  
  `date_start_prem` date NOT NULL,  
  `status_prem` tinyint(1) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `quit` (  
  `id_quit` int(11) NOT NULL,  
  `name` varchar(64) NOT NULL,
```

Продолжение приложения А

```
  `surname` varchar(64) NOT NULL,  
  `patronymic` varchar(64) NOT NULL,  
  `date_quit` date NOT NULL,  
  `status_quit` tinyint(1) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `roles` (  
  `id_roles` int(11) NOT NULL,  
  `name_roles` varchar(64) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `signup` (  
  `id` int(6) UNSIGNED NOT NULL,  
  `email` varchar(30) NOT NULL,  
  `hash` varchar(48) NOT NULL,  
  `registration_date` timestamp NOT NULL DEFAULT current_timestamp() ON  
UPDATE current_timestamp(),  
  `status` varchar(20) DEFAULT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```



```
CREATE TABLE `statuses_list` (  
  `id_statuses` int(11) NOT NULL,  
  `name_statuses` varchar(64) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `statuses_tasks` (  
  `id_tasks` int(11) NOT NULL,  
  `id_tasks_users` int(11) NOT NULL,  
  `id_tasks_sender` int(11) NOT NULL,  
  `id_statuses` int(11) NOT NULL DEFAULT 4,  
  `date_change` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
current_timestamp()  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `tasks` (  
  `id_tasks_1` int(11) NOT NULL,  
  `name_tasks` varchar(64) NOT NULL,  
  `id_sender` int(11) NOT NULL,  
  `id_receiver` int(11) NOT NULL,  
  `date_create` timestamp NOT NULL DEFAULT current_timestamp() ON UPDATE  
current_timestamp(),  
  `dead_line` date NOT NULL,  
  `description` text NOT NULL,  
  `name_file` varchar(64) NOT NULL DEFAULT 'Нет файла'  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `tp_docs` (  
  `id_doc` int(11) NOT NULL,  
  `date` date NOT NULL,  
  `name_doc` varchar(64) NOT NULL,  
  `supplier` varchar(64) NOT NULL,  
  `uuid` varchar(64) NOT NULL,  
  `nomenc` varchar(64) NOT NULL,  
  `unit` varchar(32) NOT NULL,  
  `count` int(11) NOT NULL,  
  `price` int(11) NOT NULL,  
  `stoim` int(11) NOT NULL,
```

Продолжение приложения А

```
`nds_stavka` int(11) NOT NULL,  
`nds_sum` float NOT NULL,
```

```
`excise_stavka` int(11) NOT NULL,  
`inn` varchar(64) NOT NULL,  
`rnn` varchar(64) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `tr_docs` (  
  `id_tr` int(11) NOT NULL,  
  `date` date NOT NULL,  
  `name_doc` varchar(64) NOT NULL,  
  `supplier` varchar(64) NOT NULL,  
  `receiver` varchar(64) NOT NULL,  
  `uuid` varchar(64) NOT NULL,  
  `nomenc` varchar(32) NOT NULL,  
  `unit` varchar(16) NOT NULL,  
  `count` int(11) NOT NULL,  
  `price` int(11) NOT NULL,  
  `stoim` int(11) NOT NULL,  
  `nds_stavka` int(11) NOT NULL,  
  `nds_sum` float NOT NULL,  
  `stoim_vsego` int(11) NOT NULL,  
  `excise_stavka` int(11) NOT NULL,  
  `excise_sum` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

```
CREATE TABLE `users` (  
  `id` int(11) NOT NULL,  
  `email` varchar(64) NOT NULL,  
  `password` varchar(64) NOT NULL,  
  `tel` varchar(20) NOT NULL,  
  `name` varchar(64) NOT NULL,  
  `surname` varchar(64) NOT NULL,  
  `patronymic` varchar(64) NOT NULL,  
  `gender` varchar(64) NOT NULL,  
  `birthday` date NOT NULL,  
  `location` varchar(64) NOT NULL,  
  `id_roles` int(11) NOT NULL  
) ENGINE=InnoDB DEFAULT CHARSET=utf8;
```

Приложение Б (обязательное)
Акт внедрения

Утверждаю
Директор ИП "Саржигитов
Х.А."
Саржигитов Х.А.
«20» мая 2020г.

АКТ ВНЕДРЕНИЯ

Настоящий акт составлен о том, что результаты выпускной работы студента НАО АУЭС им. Гумарбека Даукеева гр. ВТ-16-2 очной формы обучения Бобровского П.А. на тему «Разработка информационного портала для компании. Разработка системы хранения и обработки цифровых данных» внедрены в ИП "Саржигитов Х.А." и используются в компании для хранения и обработки цифровых данных. Использование результата выпускной работы Бобровского П.А. обеспечивает хранение актуальной информации о действующих сотрудниках, хранение разного рода документов (на принятие на работу, увольнение, отпуск и премию), хранение и отслеживание текущих статусов поставленных задач и их самих, хранение курса валют, а также хранение сообщений из внутреннего чата в текстовом виде.

Директор ИП "Саржигитов Х.А."



Саржигитов Х.А.

Исполнитель

Бобровский

Бобровский П.А.