

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

кафедра компьютерных технологий

«Допущен к защите»
Заведующий кафедрой _____

(Ф.И.О., ученая степень, звание)

« _____ » 20__ г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Разработка базы данных для мобильного решения
консультации клиентов компьютерного магазина

Специальность 58070400

Выполнил (а) Смагул С.С. ВТ-12-3
(Фамилия и инициалы) группа

Научный руководитель Ахметова М.А., к.т.н., доцент
(Фамилия и инициалы, ученая степень, звание)

Консультанты:

по экономической части:

Бекенбаева А.И., к.э.н., доцент
(Фамилия и инициалы, ученая степень, звание)
А.И. « 03 » 05 2016 г.
(подпись)

по безопасности жизнедеятельности:

Трихонков И.Г., д.х.н., профессор
(Фамилия и инициалы, ученая степень, звание)
И.Г. « 28 » 04 2016 г.
(подпись)

по применению вычислительной техники:

Ахметова М.А., к.т.н., доцент
(Фамилия и инициалы, ученая степень, звание)
М.А. « _____ » 20__ г.
(подпись)

Нормоконтролер: Ахметова М.А., к.т.н., доцент
(Фамилия и инициалы, ученая степень, звание)
М.А. « _____ » 20__ г.
(подпись)

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

Алматы 2016 г.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Факультет Аэрокосмических и Информационных Технологий
Специальность Вычислительная Техника и Программное Обеспечение
Кафедра Компьютерных Технологий

ЗАДАНИЕ

на выполнение дипломного проекта

Студент Синагул Султан Сарсенұлы
(фамилия, имя, отчество)

Тема проекта Разработка Базы данных для мобильного решения
консультаций клиентов компьютерного магазина

утверждена приказом ректора № 21 от «10» марта 2016 г.

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

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

Разработка Базы данных

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

Анализ предметной области

Постановка задач

Проектирование Базы данных

Установка локального сервера

Технико-экономическое обоснование

Безопасности жизнедеятельности

Перечень графического материала (с точным указанием обязательных чертежей)

Модель данных

Диаграмма компонентов

Диаграмма прецедентов

Диаграмма классов

Диаграмма развертывания

Рекомендуемая основная литература

Эрик Р. Кейбург, Роберт А. Макмизик : Проектирование баз данных с помощью UML

Кристиан Утц : PHP и MySQL

Линн Бейли, Майк Моррисон : Изучаем PHP и MySQL

Бекмусов А.И. : Методическое указание к выполнению экономической части дипломной работы.

Белов С.В., Ивашкина А.В., Козьмов А.Ф. : Безопасность информационных систем

Консультанты по проекту с указанием относящихся к ним разделов

Раздел	Консультант	Сроки	Подпись
БД	Дрихтовский И.Г.	01.04 - 28.04.16	[Подпись]
Эконом. часть	Бекмусов А.И.	01.04 - 03.05.16	[Подпись]
Предметная область	Ахметова М.А.	24.02.16 - 10.03.16	[Подпись]
Теоретическая часть	Ахметова М.А.	07.03.16 - 25.03.16	[Подпись]
Проектирование	Ахметова М.А.	24.03.16 - 13.05.16	[Подпись]
Контроль	Ахметова М.А.	20.05.16 - 30.05.16	[Подпись]

Г Р А Ф И К
подготовки дипломного проекта

№ п/п	Наименование разделов, перечень разрабатываемых вопросов	Сроки представления руководителю	Примечание
1	Анализ предметной области	24.02.16 - 10.03.16	
2	UML диаграммы	11.03.16 - 18.03.16	
3	Инструменты разработки	23.03.16 - 06.04.16	
4	Установка локального сервера	04.04.16 - 18.04.16	
5	Проектирование БД	24.03.16 - 13.05.16	

Дата выдачи задания « 16 » сентября 20 15 г.

Заведующий кафедрой _____
(подпись) (Фамилия и инициалы)

Руководитель _____
(подпись) (Фамилия и инициалы)

Задание принял к исполнению студент _____
Снаф- (подпись) Смагун С.С. (Фамилия и инициалы)

Аннотация

Тема дипломного проекта «Разработка базы данных для мобильного решения консультации клиентов компьютерного магазина».

Целью разработанной базы данных является хранение в себе всех записей по тематике проекта. База данных прежде всего необходима для связки с базой знаний, которая в последствии генерирует правила на основе знаний эксперта.

Данная база данных является реляционной, легко масштабируемой и адаптируемая ко многим разработкам.

Annotation

The theme of diploma project “The development of a database system of a mobile solution for customer consultation of a computer store”.

Purpose of developed database is storing all records of the topics of the project. The database is primarily needed to bind to the knowledge base, which subsequently generates rules based on knowledge expert.

This database is a relational, easily scalable and adaptable to many developments.

Андатпа

Дипломдық жоба тақырыбы: «Компьютерлік дүкеннің тұтынушыларына кеңес берудің мобильдік шешемі үшін дерктер қорын құру».

Деректер қорын өңдеу мақсаты, жобаның тақырыбы бойынша барлық деректерді сақтауды қамтамасыз ету. Дерекқор, ең алдымен, эксперттің біліміне сәйкес білім базасындағы ережелердің туындауы үшін қолданылып, байланыстырылған.

Бұл дерекқор оңай және бейімделгіш болып табылады.

Содержание

Введение	7
1 Планирование и начальная разработка	8
1.1 Анализ предметной области	8
1.2 Аналоги	8
1.3 Постановка задачи	11
1.4 UML диаграммы	12
1.4.1 Диаграмма компонентов	12
1.4.2 Диаграмма прецедентов	12
1.4.3 Диаграмма классов	13
1.4.4 Диаграмма развертывания	14
2 Обоснование выбора программно-аппаратных средств	15
2.1 Инструменты для проектирования БД	15
2.1.1 Erwin Data Modeler	15
2.1.2 Rational Rose	15
2.2 Платформа для СУБД	16
2.3 СУБД MySQL	20
2.3.1 Сравнение с другими СУБД	22
3 Практическая часть	25
3.1 Установка локального сервера	25
3.2 Проектирование БД	30
3.2.1 Модель данных	30
3.2.2 Назначение сущностей предметной области	31
3.2.3 Описание атрибутов сущностей	32
3.3 Разработка пользовательского интерфейса	39
4 Технико-экономическое обоснование	40
4.1 Описание работы и обоснование необходимости	40
4.2 Расчет затрат на разработку информационных технологий	40
4.3 Расчет цены программного продукта	48
4.4 Оценка социально-экономических результатов	49
5 Безопасность жизнедеятельности	50
5.1 Анализ условий труда	50
5.2 План помещения	51
5.3 Характеристики используемого оборудования	52
5.4 Расчет мер защиты от поражения электрическим током	52
5.5 Расчет системы кондиционирования	57
Заключение	63
Список литературы	64
Приложение А	65
Приложение Б	74

Введение

В нынешние дни любая сфера инфраструктуры используют автоматизированные системы для хранения, управления данными. Разработанные на данный момент технологии хранения позволяют содержать в себе огромное количество записей. Основным назначением данной технологии является обеспечение надежного хранения.

Что же понимают под названием технологии хранения? Данные технологии именуется как Система Управления Базами Данных. Под базой данных подразумевают хранилище, где данные располагаются структурированно, при этом они должны быть целостны, минимально избыточны и непротиворечивы.

На рынке Казахстана, в магазинах по продажам электронно-бытовой техники в большинстве преобладает реляционная база данных, так как она является масштабируемым, удобным в физической реализации на ЭВМ и более эффективная в обработке данных.

Целью дипломного проекта является разработка базы данных для магазина, которая занимается продажей компьютеров и его периферии. В каждом магазине существует своя база данных, которая ведет учет товаров, в частности поступления или продажи. Все данные о товарах, вплоть до детальной характеристики, содержатся в виде записей в базе данных.

В базе данных вместе с подробными характеристиками устройства, существуют определенные важные поля, которые способствуют выборки устройства, по предпочтению покупателя. Именно эти поля играют важную роль в ходе расчетов в базе знаний. База знаний на данном этапе выполняет выборку товаров и определяет правила по которым осуществляется их дальнейшее отображение на мобильном телефоне.

Для выполнения поставленной цели проекта, предусматривается сперва поставить на выполнения задачи, а именно: изучить предметную область сферы магазинов по продаже компьютерной техники, сделать обзор и анализ аналогов, выбрать инструментарий для разработки, спроектировать базу данных по тематике проекта.

1 Планирование и начальная разработка

1.1 Анализ предметной области

Магазины продажи электронной техники имеют своё направление в сфере оказания услуг. В данных магазинах содержатся товары потребления (бытового происхождения), в данном случае рассматривается техника в виде компьютеров и ноутбуков, которые необходимы людям для удовлетворения своих потребностей. У каждого товара есть свои характеристики, которые имеют отличительную особенность от остальных товаров. У каждого человека есть свои предпочтения к технике, в данном случае человек, он сам и клиент магазина, в праве выбрать сам подходящую для него устройство по типу устройства и по его бюджету.

База данных дипломного проекта разрабатывается для мобильного приложения, которое позволит человеку-клиенту совершать покупки техники дистанционно. Главной особенностью приложения будет наличие базы знаний, которая будет осуществлять помощь с выбором техники клиентам, не разбирающиеся в компьютерной технике. База знаний будет отбирать нужную технику клиентам методом эвристического опроса. По завершении опроса мобильное приложение выведет на экране клиенту нужную ему технику.

База данных содержит в себе структурированные данные о компьютерной технике, данные о клиенте и его заказе. На основе данных можно отследить какой товар по какому заказу идет к клиенту.

К базовым сущностям относятся:

- Клиенты – сущность, хранящая в себе данные о клиенте;
- Ноутбуки, моноблоки, компьютеры, периферия – сущности, в которых хранятся основные и необходимые данные о технике;
- Комплектующие компьютера и устройства, такие как: процессор, оперативная память, видеокарта, жесткий диск и т.д. – сущности которые содержат в себе характеристики о комплектующих.

1.2 Аналоги

Первым сравниваемым аналогом будет мобильное приложение магазина электроники и бытовой техники Казахстана – «Технодом».



Рисунок 1.1 – Логотип компании «Технодом»

Мобильное приложение содержит в себе все категории продаваемых товаров в магазине. Так же к каждому товару присваивается своё изображение и характеристика. Характеристика товара делится на короткую и подробную. Можно осуществить покупку товара, зарегистрировавшись в системе.

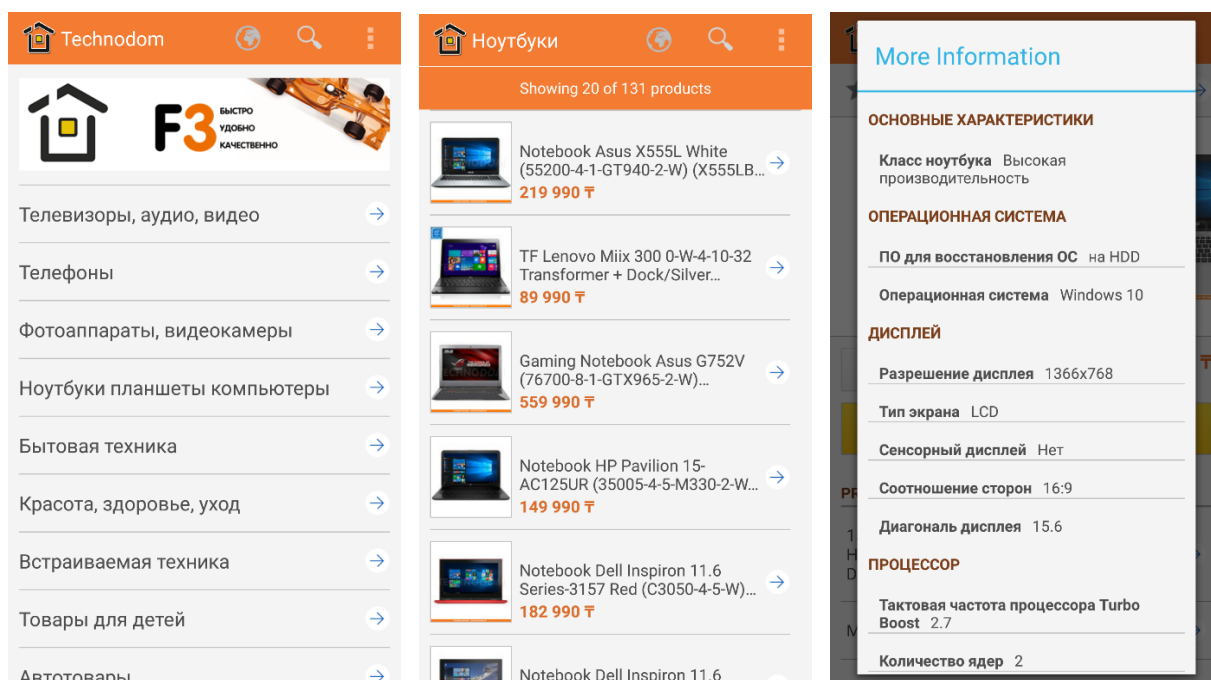


Рисунок 1.2 – Вид мобильного приложения

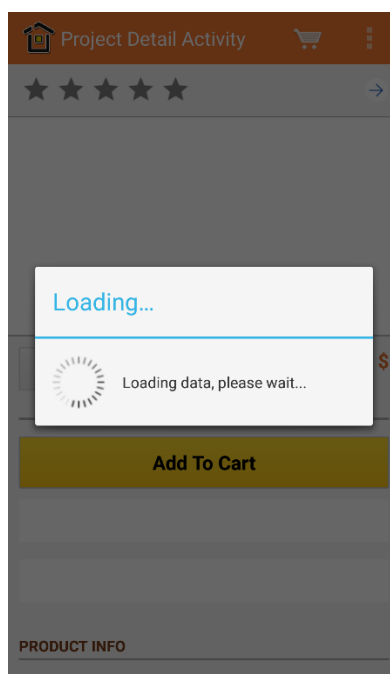


Рисунок 1.3 – Загрузка данных в мобильном приложении

Единственным отталкивающим фактором можно считать то, что мобильное приложение не оптимизировано под «слабые» телефоны. Любые данные, то есть, список товаров, изображения и их описания, мобильное приложение берет с сайта, что собственно немного подгружает мобильное устройство.

Следующим аналогом послужит мобильное приложение под названием «Buying Guide».

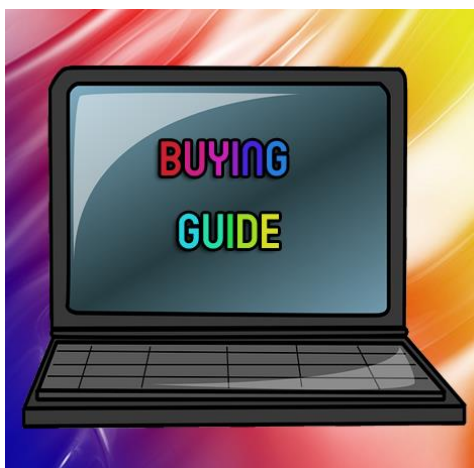


Рисунок 1.4 – Логотип мобильного приложения «Buying Guide»

Данное мобильное приложение является справочником (руководством), которое взаимодействуя с пользователем, инструктирует его пошагово, помогая выбирать устройство. Мобильное приложение не имеет в составе интернет-магазина, в котором осуществлялась бы продажа.

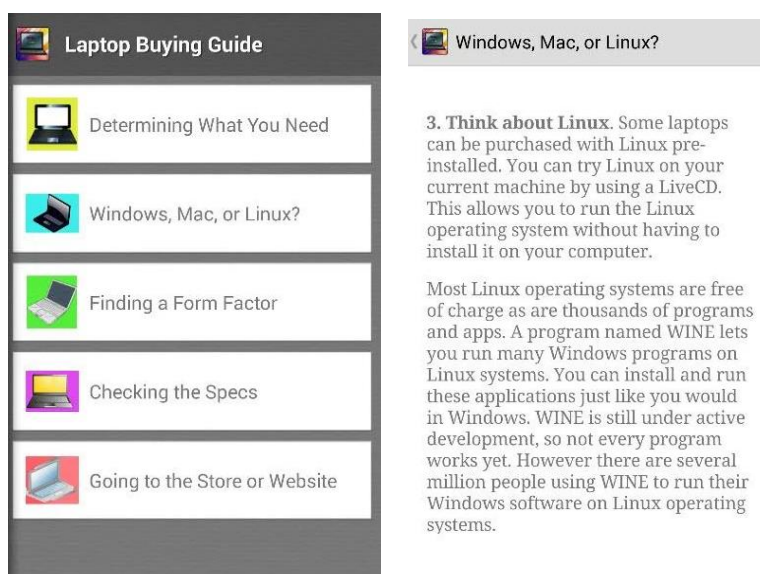


Рисунок 1.5 – Вид мобильного приложения

Следующим аналогом будет наименее популярный Казахстанский интернет-магазин – «Disti».



Рисунок 1.6 – Логотип интернет-магазина

Интернет-магазин занимается продажей компьютерной и бытовой техники. Имеется возможность приобретать товар, изначально зарегистрировавшись. Данный интернет-магазин не имеет мобильного приложения что делает покупки неудобными.

1.3 Постановка задачи

Задачами дипломного проекта является создание базы данных для магазина продажи электронной техники. Ориентир базы данных будет на товары электронной техники, а именно: настольные компьютеры, моноблоки, ноутбуки и периферия. База данных должна фиксировать товары, заказанные клиентом.

База данных содержит следующие возможности:

- Хранение данных о клиентах;
- Хранение данных о комплектующих компьютера;
- Хранение данных о производителях;
- Хранение данных о заказах клиентов;
- Хранение данных об устройствах;
- Хранение данных о комплектующих устройства;
- Хранение данных о периферии.

База данных рассчитана на следующую работу:

- Регистрация новых клиентов в системе;
- Пополнение товаров.

Администратор имеет следующие возможности:

- Редактирование всех таблиц;
- Заполнение, изменение, удаление полей таблиц.

Клиент взаимодействует с таблицами следующим образом:

- Просмотр интересующих его товаров;
- Получение сведения о товаре;
- Заказ товара.

К базе данных будет разработан интерфейс, который будет способствовать заполнению и удалению записей из базы. Интерфейс предназначен для людей, которые не разбираются в запросах СУБД.

1.4 UML диаграммы

1.4.1 Диаграмма компонентов

Диаграмма компонентов рассматривает особенности физического представления и отношения между ними. Физическим типом здесь рассматривается сам компонент, так как оно является заменяемой частью системы и имеет набор интерфейсов. [2]

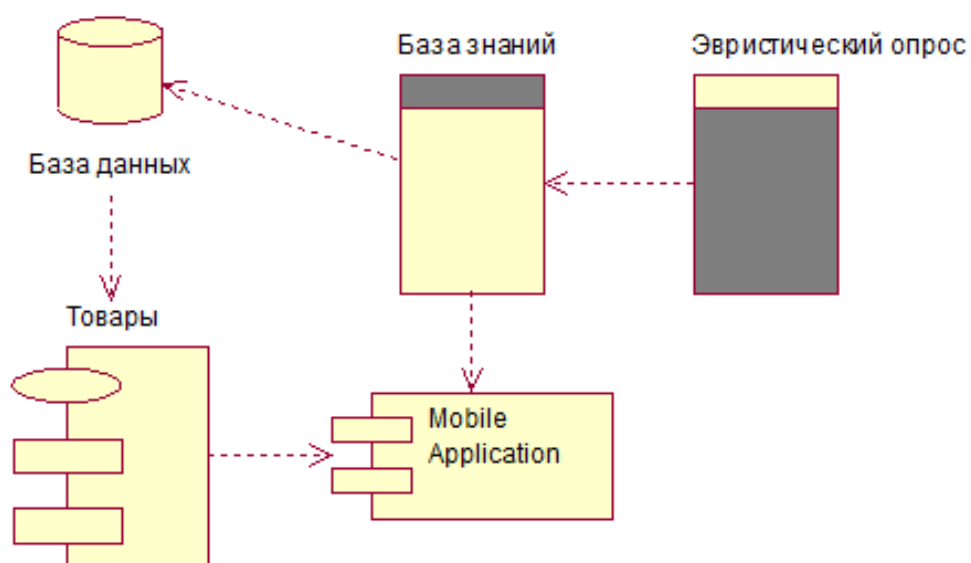


Рисунок 1.7 – Диаграмма компонентов

1.4.2 Диаграмма прецедентов

Диаграмма прецедентов показывает взаимодействие актеров (пользователей) с базой данных. [2]

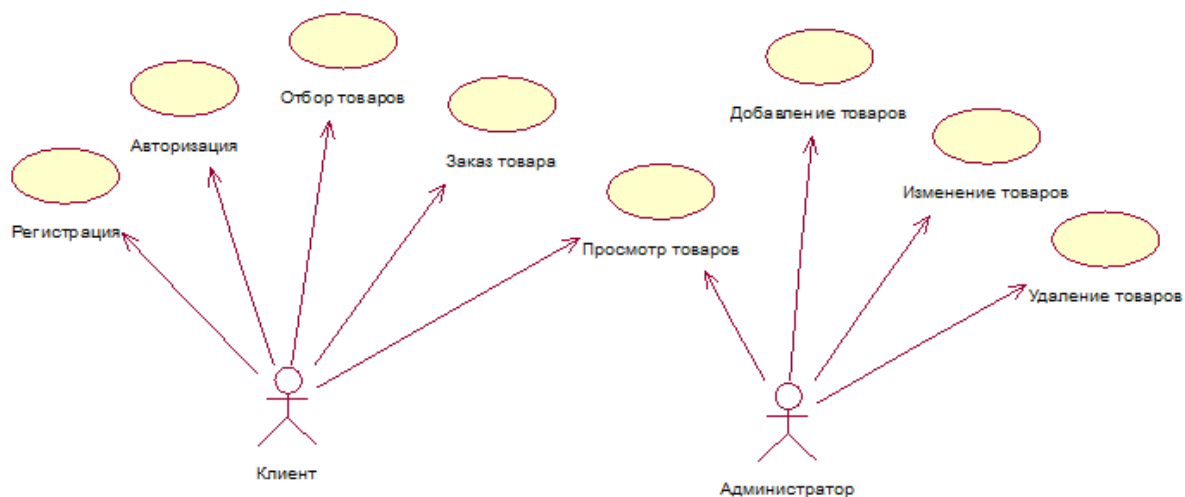


Рисунок 1.8 – Диаграмма прецедентов

1.4.3 Диаграмма классов

Диаграмма классов описывает статическую структуру системы, где применяются только один тип сущностей – это классы. Диаграмма классов может быть частным случаем ER диаграммы, так как в UML больше уделяется внимание поведению класса. [4]

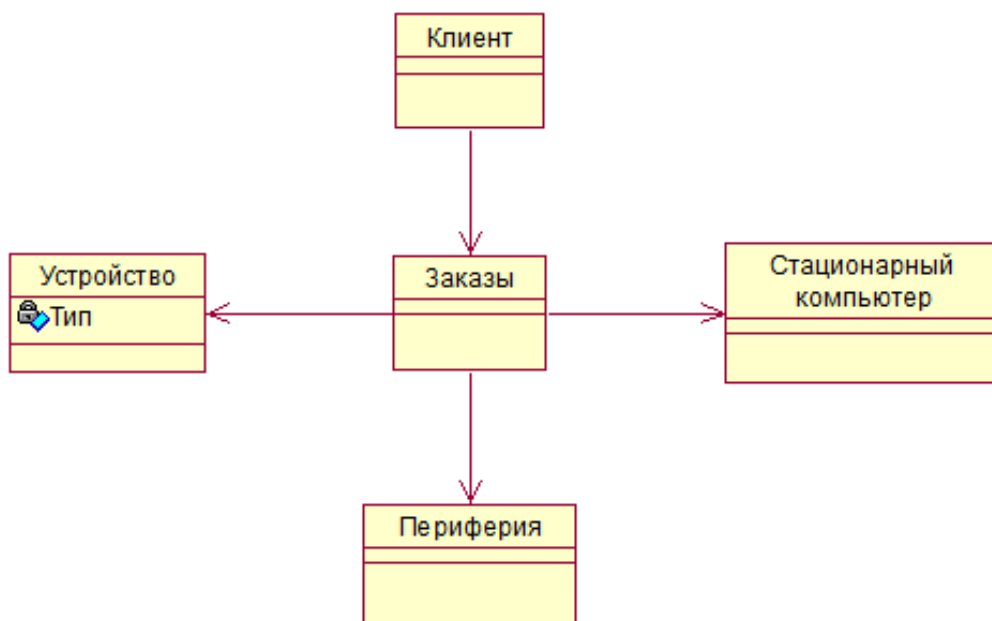


Рисунок 1.9 – Диаграмма классов

1.4.4 Диаграмма развертывания

Диаграмма развертывания демонстрирует взаимодействие системы в целом. [4]

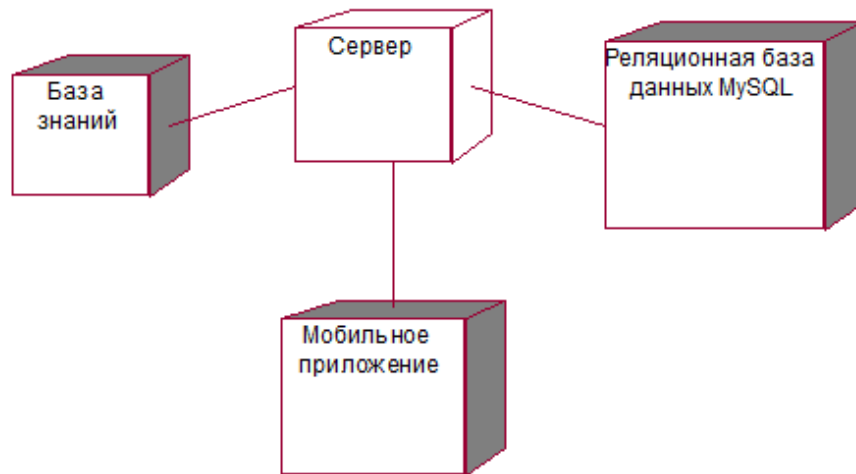


Рисунок 1.10 – Диаграмма развертывания

2 Обоснование выбора программно-аппаратных средств

2.1 Инструменты для проектирования БД

Разработка и проектирование базы данных всегда требует стороннего вмешательства программных обеспечений, которые могут помочь в проектировании добавив главную черту – визуальность и автоматизированность. Данными свойствами обладают два программных обеспечения – ERwin и Rational Rose.

2.1.1 Erwin Data Modeler

Erwin Data Modeler – CASE-средство, помогающая пользователю-разработчику заниматься проектированием, документированием и сопровождением базы данных. Приложение является передовым решением для моделирования базы данных, что в следствии помогает визуально отобразить его для четкого представления и понимания. Визуальность обеспечивает эффективно организовать процесс, управление и администрирование внутри любого предприятия где используются технологии баз данных.

Данный продукт разработан компанией Logic Works, которая выпустила программы ERwin, BPwin и OOwin.

Erwin Data Modeler является автоматизированной системой, которая решает множество трудоемких задач, что в следствии уменьшит сроки создания баз данных и хранилищ.

2.1.2 Rational Rose

У каждого разработчика есть свои идеи в разработке чего-либо. Иметь общий язык моделирования и одновременно простой в использовании инструмент, который может документировать и уточнить эти идеи в артефакты, которые могут быть переданы для повторного использования, в общности можно назвать это шаблоном.

Семейство продукта IBM Rational Rose позволит пользователям разрабатывать программное обеспечение с помощью унифицированного языка моделирования (UML). Rational Rose может совмещать в себе архитектуру, аналитику, программное обеспечение и визуальное представление базы данных, которые могут использовать разработчики для построения визуальных моделей для разработки проектов.

С помощью промышленного стандарта UML, пользователи могут создавать семантически богатые архитектуры приложений, которые помогут команде разработчиков, в том числе заинтересованной стороне, разработать четкое понимание проекта. Общий инструмент и язык для моделирования

помогает упростить рабочую среду и ускорить развитие качества программного приложения. [4]

Главными особенностями данного продукта можно выделить:

- Помощь командам создания «богатого» приложения на основе надежной архитектуры;
- Позволяет пользователям создавать платформу вне зависимости от архитектуры с полной поддержкой UML 1.x;
- Сопровождает общий инструмент моделирования и язык для использования по всей команде разработчиков;
- Проектирование по языкам Ada, ANSI C++, C, C++, CORBA, Java, Microsoft Visual Basic и Microsoft Visual C++, также моделирование базы данных;
- Поддержка в режиме реального времени и разработка встраиваемых систем;
- Вмещает хорошо известные шаблоны для ANSI C ++, Java и код Visual C++;
- Поддержка Enterprise Java Beans и веб-моделирование;
- Включает веб-публикации и отчетность.

Продукты Rational Rose включают в себя следующие типы UML диаграмм: активности, классов, компонентов, развертывания, состояния, последовательности.

Программное обеспечение IBM Rational Rose Enterprise обеспечивает поддержку моделирования для применения разработки и работы с рядом технологий реализации. Все продукты Rational Rose включают в себя полную поддержку UML, однако, они различаются с точки зрения реализации технологиями, которые они поддерживают.

2.2 Платформа для СУБД

Linux является свободным и открытым исходным кодом операционной системы, разработанная Линусом Торвальдсом и вступил в существование в 1991 году, выпущенный под GPL (General Public License). Как и любая другая ОС, основной функцией ОС Linux является управление системными ресурсами (Hardware). GNU Linux получил много успеха и популярности в последние 2 десятилетия, большинство коммерческих серверов в настоящее время используют GNU Linux. В последнее время домашние пользователи также начали использовать Linux в качестве повседневной ОС, и популярные ноутбуки и ПК также используют GNU Linux в качестве предустановленной ОС на своих системах, так как его свободная лицензия помогает в сокращении общих затрат. [9]

Операционная система Linux использует многоуровневую архитектуру, которая состоит из следующих слоев:

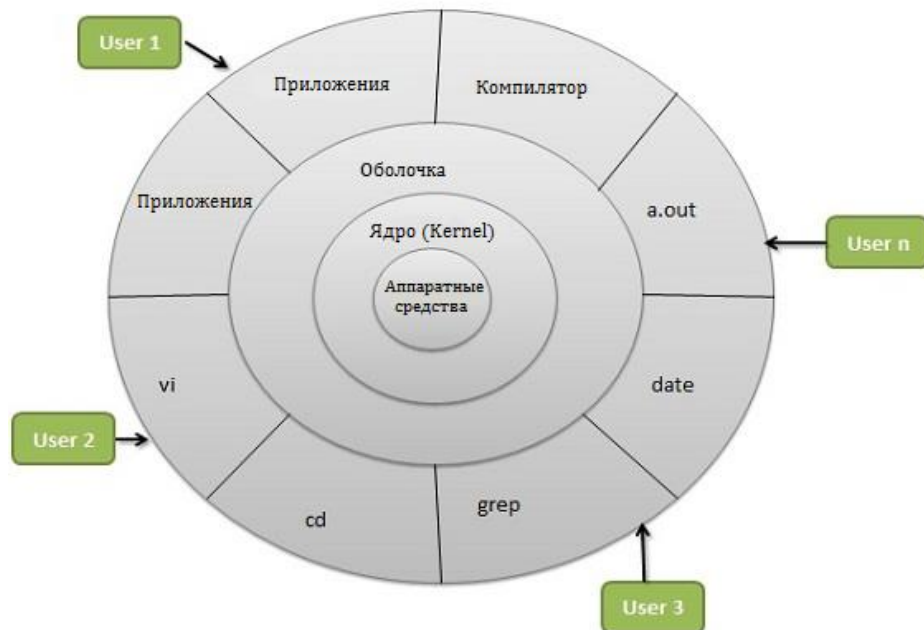


Рисунок 2.1 – Многоуровневая архитектура ОС Linux

- Слой аппаратного средства – оборудование состоит из всех периферийных устройств (RAM / HDD / CPU и т.д.);
- Ядро – основной компонент операционной системы, для всех основных управления ввода/вывода он взаимодействует непосредственно с аппаратными средствами;
- Оболочка – интерфейс между пользователем и ядром, скрывающей сложность функций ядра так же от пользователей. Принимает команду от пользователя и выполняет функции ядра;
- Программное обеспечение приложения – программы, дающие пользователю большую часть функциональности операционной системы;
- Пользователи – пользователи системы, которые взаимодействуют непосредственно с системой и приложениями программного обеспечения.

Linux является быстрым, бесплатным и простым в использовании для мощных ноутбуков и серверов по всему миру. Linux имеет гораздо больше возможностей удивлять своих пользователей, таких как:

Live CD / USB: Почти все дистрибутивы Linux имеют функцию Live CD / USB, с помощью которого пользователь может запускать / попробовать ОС даже не устанавливая его в систему.

Графический пользовательский интерфейс (X Window System). Люди думают, что Linux является операционной системой где преобладает только

командная строка, но всё же существуют такие дистрибутивы, но нельзя предполагать, что все Linux-подобные системы такие. В Linux есть установочные пакеты, которые могут преобразить графический интерфейс на подобии Windows. Такие пакеты именуется графическими оболочками и к ним можно соотнести: Gnome, KDE, Xfce, MATE и другие.

Поддержка большинства языков. Linux используется во всем мире и, следовательно, доступна на многих языках, и поддерживает большинство своих пользовательских национальных клавиатур.

Поддержка приложений. Linux имеет свой собственный репозиторий программного обеспечения, из которого пользователи могут загружать и устанавливать тысячи приложений только путем выдачи команды в Linux Terminal или Shell. Linux также может запускать приложения Windows, если это необходимо. [9]

Linux имеет несколько уникальных характерных черт, некоторые из которых являются наиболее важными:

- Многозадачность. Linux имеет возможность обрабатывать более одного задания в то время, скажем, например, вы выполнили команду для сортировки для огромного списка и одновременно вводите в блокноте. Это управляется путем деления процессорного времени разумно путем реализации политики планирования и концепции переключения контекста.

- Переносимость. Переносимость была одной из главных особенностей, которые сделали Linux настолько популярным среди пользователей, но портативность не означает, что он меньше по размеру файла и могут быть перенесены на флэш-накопитель, компакт-дисков и карт памяти. Вместо этого здесь портативность означает, что операционная система Linux и ее применение может работать на различных видах оборудования таким же образом. Ядро Linux и прикладные программы поддерживают их установку даже на очень минимальных настройках аппаратной конфигурации.

- Многопользовательские возможности. Это способность ОС Linux, где одни и те же ресурсы компьютера - жесткий диск, память и т.д. доступны для нескольких пользователей. Конечно, не на одном терминале, они дают разные терминалы, чтобы работать с системой. Все терминалы затем подключаются к главному серверу или хост-машине Linux, чьи ресурсы и подключенные периферийные устройства, такие как принтер, могут быть использованы. Примером многопользовательских возможностей может послужить архитектура «клиент / сервер», где различные клиенты подключены к серверу Linux. Клиент отправляет запрос на сервер с определенными запросами данных, а сервер обрабатывает входящие данные запрашиваемого файла.

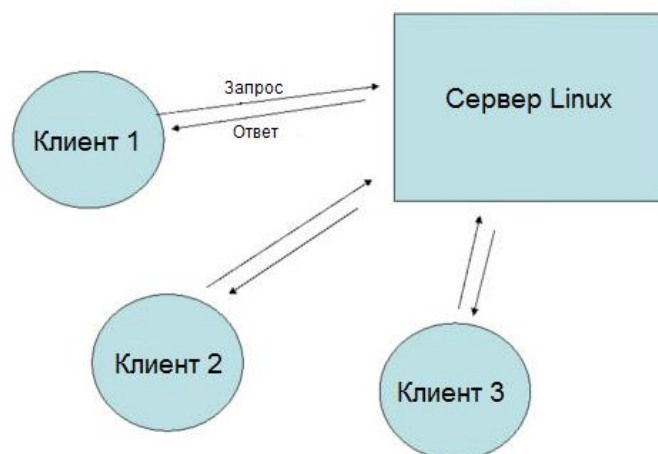


Рисунок 2.2 – Архитектура клиент / сервер

– Безопасность. Безопасность является очень важной частью любой операционной системы, для организации / пользователя, который использует систему для своих секретных работ, Linux действительно обеспечивает несколько концепций безопасности для защиты своих пользователей от несанкционированного доступа своих данных и системы. Linux обеспечивают 3 основные концепции безопасности:

1) Аутентификация. Это значит, каждый зарегистрированный человек в системе должен идентифицировать себя путем ввода логина и соответствующего для него пароля, которые должны быть известны только владельцу, чтобы предотвратить несанкционированный доступ к своему логину.

2) Авторизация. На уровне файла Linux имеет пределы авторизации для пользователей, есть чтение, запись и выполнение для каждого файла, который решает, кто может получить доступ к какой-либо файл, который может изменить его, и кто может его выполнить.

3) Шифрование. Эта функция кодирует файлы в нечитаемый формат, который также известен как "Шифротекст", так что даже если кто-то сможет открыть файл, то все данные в нем, то есть символы будут нечитаемыми что сохранит безопасность файла.

– Связь. Linux имеет отличную возможность для общения с другими пользователями, он может находиться в пределах сети одного основного компьютера, или между двумя или более таких компьютерных сетей. Пользователи могут легко обмениваться почтой, данные программы с помощью таких сетей.

2.3 СУБД MySQL

MySQL – популярная и распространенная система управления базами данных (СУБД), используемая в сочетании с PHP. База данных (БД) имеет совокупность структурированных данных и имеет реляционный тип. Реляционная БД позволяет связывать между собой таблицы при помощи отношений, которые в последствии объединятся в одну общую, структурированную таблицу. [8]

Основные особенности MySQL: Как и большинство других баз данных, MySQL является реляционной базой данных;

MySQL является системой клиент/сервер. Существует сервер базы данных и много клиентов (прикладных программ), которые взаимодействуют с сервером; то есть, запрос данных, сохранение изменений и так далее. Клиенты могут работать на том же компьютере, что и сервер, или на другом компьютере (связи через локальную сеть или Интернет).

Совместимость с SQL. MySQL поддерживает в качестве языка базы данных – SQL (Structured Query Language). SQL представляет собой стандартизированный язык для выполнения запросов и обновления данных и для администрирования базы данных. Есть несколько диалектов SQL (примерно столько же, как есть системы управления базами данных). MySQL придерживается текущего стандарта SQL с 2003 года, хотя и со значительными ограничениями и большим количеством расширений.

Юникод. MySQL поддерживает все наборы символов.

Пользовательский интерфейс. Существует множество удобных пользовательских интерфейсов для администрирования сервера MySQL.

Поиск по всему тексту. Полнотекстовый поиск упрощает и ускоряет поиск слов, которые расположены в пределах текстового поля. Если вы используете MySQL для хранения текста (например, в интернет-дискуссионной группе), вы можете использовать полнотекстовый поиск, чтобы осуществить просто эффективную функцию поиска.

Репликации. Репликация позволяет копировать содержимое базы данных на несколько компьютеров. На практике это делается по двум причинам: для усиления защиты от сбоев системы (так что, если один компьютер выйдет из строя, другой может быть введен в эксплуатацию), а также повысить скорость запросов к базе данных.

Транзакции. В контексте системы баз данных, транзакция означает выполнение нескольких операций с базой данных в качестве блока. Система базы данных гарантирует, что-либо все операции, не выполнены правильно, или ни один из них. Это справедливо, даже если в середине транзакции происходит сбой питания, компьютер зависает, или при возникновении какой-либо другой катастрофы. Так, например, он может не произойти, что

денежная сумма снимается со счета А, но не может быть осажден в счет В из-за какой-то системной ошибки. Транзакции также дают программистам возможность прерывать серию уже выполненных команд (своего рода отзыва). Во многих случаях это приводит к значительному упрощению процесса программирования. Несмотря на общественное мнение, MySQL поддерживает транзакции в течение длительного времени. Следует отметить, что MySQL может хранить таблицы в различных форматах. Формат таблицы по умолчанию называется MyISAM, и этот формат не поддерживает транзакции. Но есть целый ряд дополнительных форматов, которые делают поддержку транзакций.

Ограничения внешнего ключа. Эти правила гарантируют, что нет никаких перекрестных ссылок в связанных таблицах, которые ведут в никуда. MySQL поддерживает ограничения внешних ключей для таблиц InnoDB.

Функции GIS (Geographic Information System). MySQL поддерживает хранения и обработки двумерных географических данных. Таким образом MySQL хорошо подходит для GIS приложений.

Языки программирования. Есть целый ряд API (интерфейсов прикладного программирования) и библиотеки для разработки приложений MySQL. Для программирования клиента можно использовать, в частности, языки C, C++, Java, Perl, PHP, Python и Tcl.

ODBC (Open Database Connectivity). MySQL поддерживает интерфейс ODBC Connector/ODBC. Это позволяет MySQL решать все обычные языки программирования, которые работают под Microsoft Windows (Delphi, Visual Basic и т.д.). Интерфейс ODBC также может быть реализован под Unix.

Независимость от платформы. Это не только клиентские приложения, которые работают под управлением различных операционных систем; Сам MySQL (то есть, сервер) может быть выполнен в соответствии с рядом операционных систем. Наиболее важными из них являются Apple Macintosh OS X, Linux, Microsoft Windows, а также бесчисленные варианты Unix, такие как AIX, BSDI, CentOS, FreeBSD, HP-UX, OpenBSD, Net BSD, SGI Iris и Sun Solaris.

Скорость. MySQL считается очень быстрой базой данных. Эта скорость была подкреплена большим количеством эталонных тестов.

Для определения альтернативных аналогов СУБД MySQL сперва нужно определить её назначения, достоинства и недостатки.

Функционал MySQL очень богат и позволяет взаимодействовать с различными сайтами и веб-приложениями. Так же он содержит в себе большой набор инструментов для разработки и создания приложения. [9]

К преимуществам MySQL можно соотнести:

- Быстрая работа – производительность данной СУБД оправдывает себя поскольку в нем упрощены некоторые стандарты что приводит его к быстрой работе;
- Легкий и доступный интерфейс – установка проходит довольно просто, а поддерживаемые им пользовательский интерфейсы (GUI) упрощают работу;
- Огромный функционал – данное СУБД поддерживает большинство функционала SQL;
- Масштабируемость – позволяет легко работать и обрабатывать большие объемы данных, легко расширяется.

Недостатки MySQL:

- Надежность – обработка данных MySQL в частности связей, транзакций и аудитов может уступать другим СУБД по надежности;
- Ограничения – из-за производительности у MySQL встроены некоторые ограничения функционала.

2.3.1 Сравнение с другими СУБД

Microsoft SQL Server – реляционная СУБД, производимая корпорацией Майкрософт. Основным языком запросов является Transact-SQL в котором реализована стандартный ANSI / ISO Structured Query Language (SQL), используемый как в Microsoft и Sybase. Microsoft SQL Server поддерживает атомарные, последовательные, изолированные, и длительные операции. Она включает в себя поддержку зеркального отображения базы данных и кластеризации. Кластер сервера SQL представляет собой набор одинаково настроенных серверов, которые помогают распределить нагрузку между несколькими серверами. SQL Server также поддерживает разделение данных для распределенных баз данных, в дополнение к зеркальному отображению базы данных которая позволяет создавать зеркала содержимого базы данных, наряду с журналами транзакций, на другом экземпляре SQL Server, на основе определенных заранее триггеров.

Oracle Database – реляционная СУБД выпущенная корпорацией Oracle, и она содержит, по меньшей мере, один экземпляр приложения, наряду с хранением данных. Экземпляр включает в себя набор процессов операционной системы и структурной памяти, которые взаимодействуют с хранилищем. В дополнение к хранению, база данных состоит из оперативных журналов транзакций, которые хранят транзакционную историю. Процессы могут в свою очередь вести оперативный журнала в архиве журналов, которые обеспечивают восстановление данных и для некоторых форм репликации данных. Хранилище данных Oracle RDBMS логически представлена в виде таблицы-пространств и физически в виде файлов данных.

На физическом уровне, файлы данных содержат один или более блоков данных, где размер блока может варьироваться от файлов данных. Особенностью Oracle являются словари данных, индексы, и кластеры. Версии последующие до 10g, включают сетевые вычислительные возможности, в которой приложение экземпляра может использовать ресурсы центрального процессора от другого узла в сетке.

DB2 является одной из ответвлений от IBM, является реляционной СУБД, которая работает на Unix, Windows или Linux серверных машинах. DB2 можно вводить либо из командной строки или из графического интерфейса. Интерфейс командной строки требует больше знаний о продукте, но может быть более легким при программировании и автоматизации. Графический интерфейс пользователя является мультиплатформенным Java-клиентом, который содержит множество мастеров, подходящих для начинающих пользователей. DB2 поддерживает как SQL и XQuery. DB2 имеет встроенную реализацию хранения данных XML для более быстрого доступа с помощью XQuery. DB2 также поддерживает интеграцию в Eclipse, и Visual Studio .NET интегрированную среду разработки. Важной особенностью DB2 СУБД является обработка ошибок, в котором структура SQL возвращает информацию об ошибке приложению.

Microsoft Office Access, ранее известный как Microsoft Access, является СУБД от Microsoft, которая сочетает в себе реляционную Microsoft Jet Database Engine с графическим пользовательским интерфейсом и инструментами разработки программного обеспечения. Он является компонентом системы Microsoft Office 2010. Одним из преимуществ доступа с точки зрения программиста является его относительная совместимость с запросами SQL. В отличие от полных реляционных СУБД, в Jet Engine не хватает триггеров баз данных и хранимых процедур. Несмотря на это, он предоставляет специальный синтаксис, который позволяет создавать запросы с параметрами, таким образом, что оно выглядит как создание хранимых процедур. Microsoft Access действительно позволяет содержать формы в котором хранится код, который срабатывает по мере внесения изменений в основную таблицу, и он является общим для использования сквозных запросов и другие методы в доступе к запуску хранимых процедур в RDBMSs, которые поддерживают их. MS Access используется в небольших предприятиях, в отделах крупных корпораций, и программистами для создания специальной системы для обработки создания и манипулирования данными. Некоторые профессиональные разработчики приложений используют Access для быстрой разработки приложений, особенно для создания прототипов и автономных приложения, которые служат в качестве инструментов.

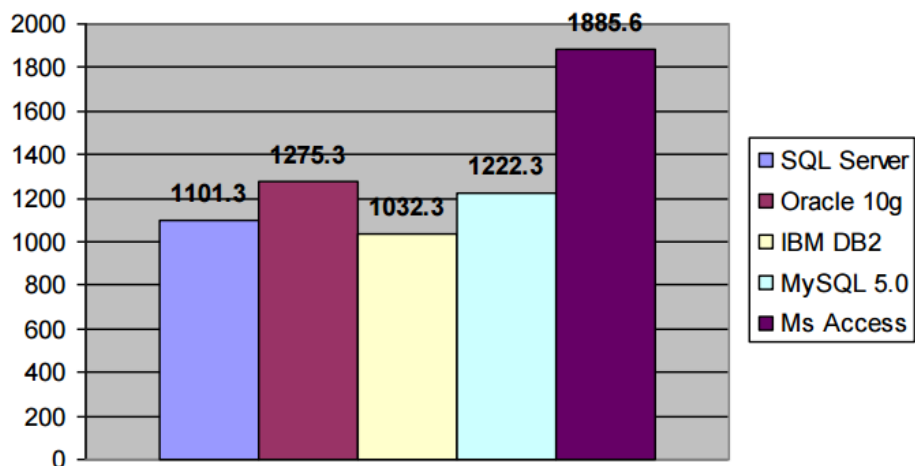


Рисунок 2.3 – Среднее время выполнения запросов

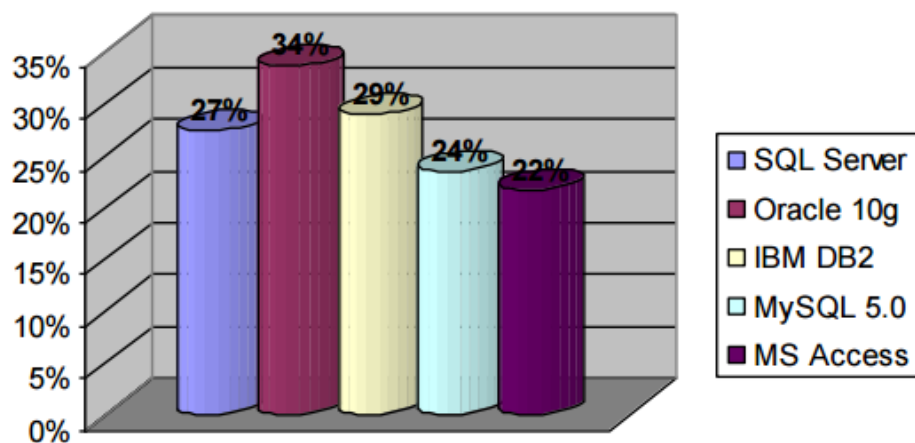


Рисунок 2.4 – Потребление центрального процессора

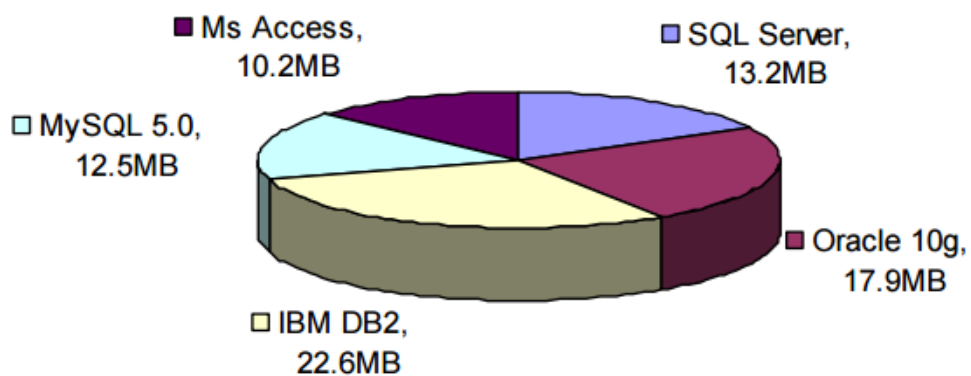


Рисунок 2.5 – Потребление оперативной памяти

3 Практическая часть

3.1 Установка локального сервера

Для содержания базы данных требуется наличие локального сервера, где будет храниться СУБД и база данных. В качестве операционной системы для локального сервера выбрана CentOS 6.5 из семейства Linux. Установка операционной системы разворачивалась на виртуальной машине с использованием программного обеспечения VMware. [9]

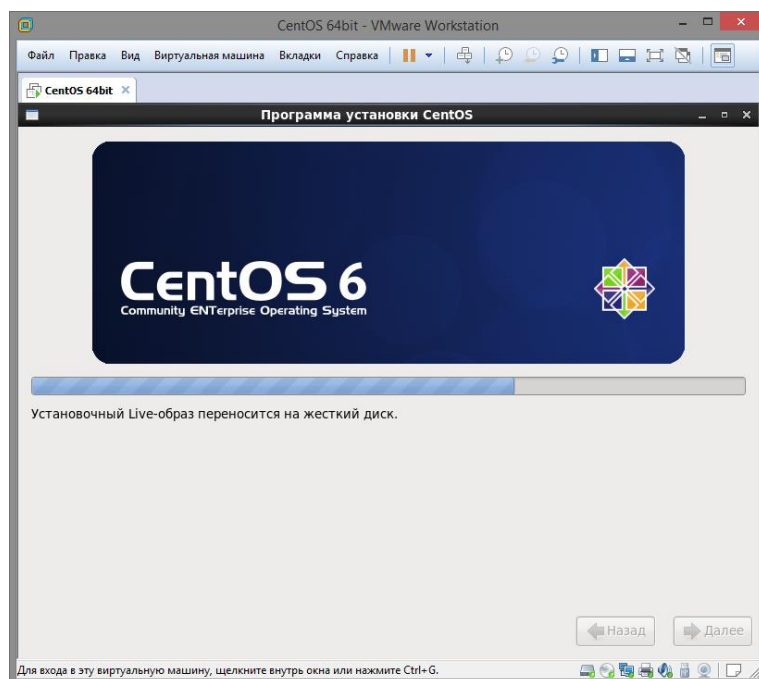


Рисунок 3.1 – Окно установки CentOS

После установки операционной системы требуется установить LAMP. Аббревиатура LAMP означает Linux, Apache, MySQL, PHP. В данном случае Apache и является локальным сервером, который будет разворачиваться на CentOS.

Первым делом установим MySQL. Установка проводится в терминале, где посредством команд, можно скачать исходники последней версии и установить их.

Команды установки следующие:

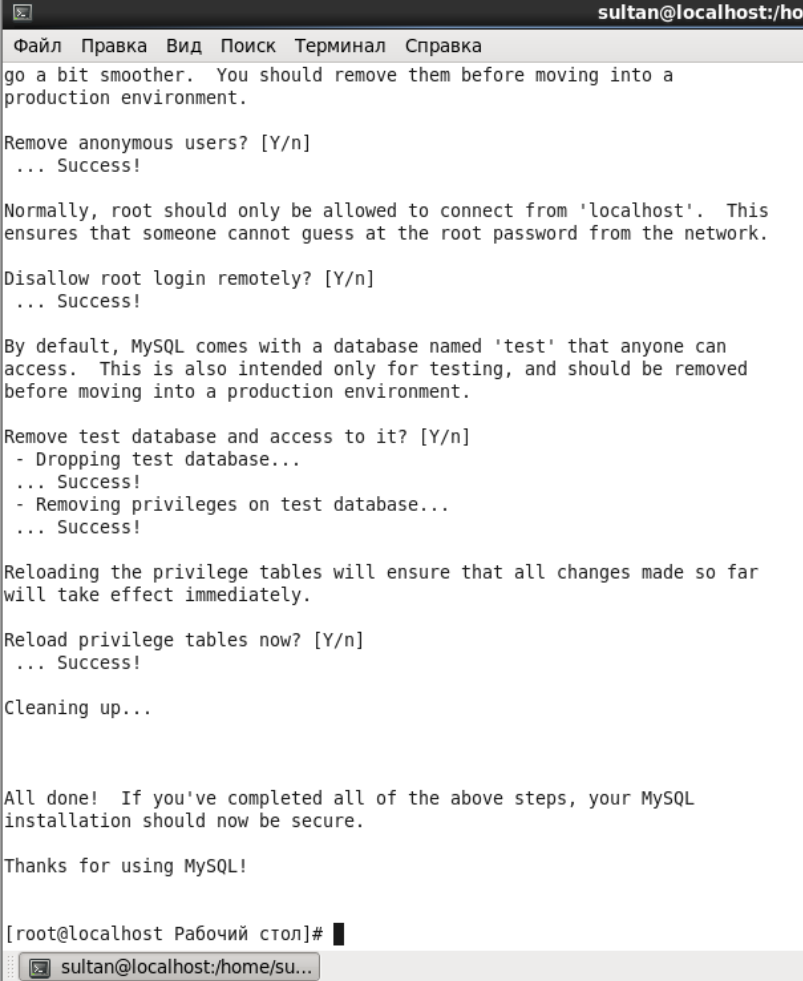
– Установка MySQL:

```
yum -y install mysql mysql-server
```

– Запуск MySQL сервера:

```
chkconfig --levels 235 mysqld on  
/etc/init.d/mysqld start
```

- Задание пароля для рута:
mysql_secure_installation



```
sultan@localhost:/ho
Файл Правка Вид Поиск Терминал Справка
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n]
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n]
... Success!

By default, MySQL comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n]
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!

Reloading the privilege tables will ensure that all changes made so far
will take effect immediately.

Reload privilege tables now? [Y/n]
... Success!

Cleaning up...

All done! If you've completed all of the above steps, your MySQL
installation should now be secure.

Thanks for using MySQL!

[root@localhost Рабочий стол]#
sultan@localhost:/home/su...
```

Рисунок 3.2 – Лог об установки MySQL

Далее устанавливается локальный сервер – Apache. Ход установки:

- Установка Apache:

```
yum -y install httpd
```

- Задание конфигурации при запуске Apache:

```
chkconfig --levels 235 httpd on
```

- Запуск Apache:

```
/etc/init.d/httpd start
```

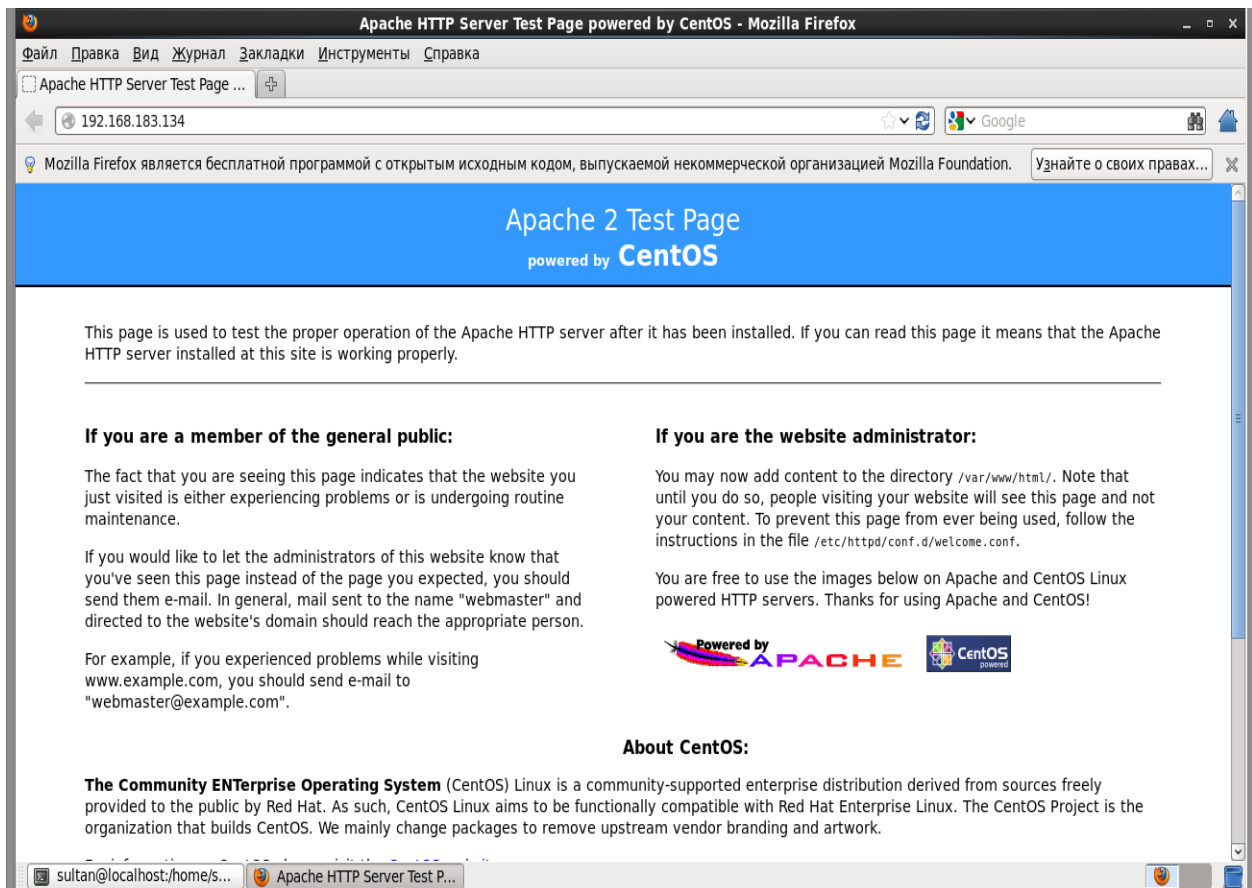


Рисунок 3.3 – Установленный Apache

Следующим компонентом для установки будет PHP. PHP будет служить связным компонентом между базой данных и локальным сервером. Ход установки:

- Установка PHP и его модулей для Apache:
yum -y install php
- Перезапуск Apache:
/etc/init.d/httpd restart

PHP Version 5.3.3	
System	Linux localhost.localdomain 2.6.32-431.el6.x86_64 #1 SMP Fri Nov 22 03:15:09 UTC 2013 x86_64
Build Date	May 10 2016 21:40:16
Configure Command	'./configure' '--build=x86_64-redhat-linux-gnu' '--host=x86_64-redhat-linux-gnu' '--target=x86_64-redhat-linux-gnu' '--program-prefix=' '--prefix=/usr' '--exec-prefix=/usr' '--bindir=/usr/bin' '--sbindir=/usr/sbin' '--sysconfdir=/etc' '--datadir=/usr/share' '--includedir=/usr/include' '--libdir=/usr/lib64' '--libexecdir=/usr/libexec' '--localstatedir=/var' '--sharedstatedir=/var/lib' '--mandir=/usr/share/man' '--infodir=/usr/share/info' '--cache-file=../config.cache' '--with-libdir=lib64' '--with-config-file-path=/etc' '--with-config-file-scan-dir=/etc/php.d' '--disable-debug' '--with-pic' '--disable-rpath' '--without-pear' '--with-bz2' '--with-exec-dir=/usr/bin' '--with-freetype-dir=/usr' '--with-png-dir=/usr' '--with-xpm-dir=/usr' '--enable-gd-native-ttf' '--without-gdgm' '--with-gettext' '--with-gmp' '--with-iconv' '--with-jpeg-dir=/usr' '--with-openssl' '--with-pcre-regex=/usr' '--with-zlib' '--with-layout=GNU' '--enable-exif' '--enable-ftp' '--enable-magic-quotes' '--enable-sockets' '--enable-syssem' '--enable-sysvshm' '--enable-sysvmsg' '--with-kerberos' '--enable-ucd-snmp-hack' '--enable-shmop' '--enable-calendar' '--without-sqlite' '--with-libxml-dir=/usr' '--enable-xml' '--with-system-tzdata' '--with-apxs2=/usr/sbin/apxs' '--without-mysql' '--without-gd' '--disable-dom' '--disable-dba' '--without-unixODBC' '--disable-pdo' '--disable-xmlreader' '--disable-xmlwriter' '--without-sqlite3' '--disable-phar' '--disable-fileinfo' '--disable-json' '--without-pspell' '--disable-wddx' '--without-curl' '--disable-posix' '--disable-sysvmsg' '--disable-sysvshm' '--disable-sysvsem'
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc

Рисунок 3.4 – Установленный PHP

Следующим шагом будет обеспечение поддержки PHP для MySQL. Потребуется установить дополнительные модули:

```
yum search php
```

– Установка модуля:

```
yum -y install php-mysql
```

```
yum -y install php-gd php-imap php-ldap php-odbc php-pear php-xml php-xmlrpc php-mbstring php-mcrypt php-mssql php-snmp php-soap php-tidy curl curl-devel
```

Настоятельно рекомендуется установить APC (Alternative PHP Code), которое позволит ускорить PHP посредством кэширования.

– Установка APC:

```
yum -y install php-pecl-apc
```

– Перезапуск Apache:

```
/etc/init.d/httpd restart
```

apc

APC Support	enabled
Version	3.1.9
APC Debugging	Disabled
MMAP Support	Enabled
MMAP File Mask	/tmp/apc.LOwPk2
Locking type	pthread mutex Locks
Serialization Support	php
Revision	\$Revision: 308812 \$
Build Date	Jun 22 2012 09:47:01

Directive	Local Value	Master Value
apc.cache_by_default	On	On
apc.canonicalize	Off	Off
apc.coredump_unmap	Off	Off
apc.enable_cli	Off	Off
apc.enabled	On	On
apc.file_md5	Off	Off
apc.file_update_protection	2	2
apc.filters	<i>no value</i>	<i>no value</i>
apc.gc_ttl	3600	3600
apc.include_once_override	Off	Off
apc.lazy_classes	Off	Off
apc.lazy_functions	Off	Off
apc.max_file_size	1M	1M
apc.mmap_file_mask	/tmp/apc.LOwPk2	/tmp/apc.LOwPk2
apc.num_files_hint	1024	1024

Рисунок 3.5 – Лог установки APC

Следует установить компонент phpMyAdmin, который позволит манипулировать базами данных MySQL через графический интерфейс.

Ход установки следующий:

– Импортирование репозитория RPMforge:

```
rpm --import http://dag.wieers.com/rpm/packages/RPM-GPG-KEY.dag.txt
```

```
yum -y install http://pkgs.repoforge.org/rpmforge-release/rpmforge-release-0.5.3-1.el6.rf.x86_64.rpm
```

– Установка phpMyAdmin:

```
yum -y install phpmyadmin
```

– Перезапуск Apache:

```
/etc/init.d/httpd restart
```

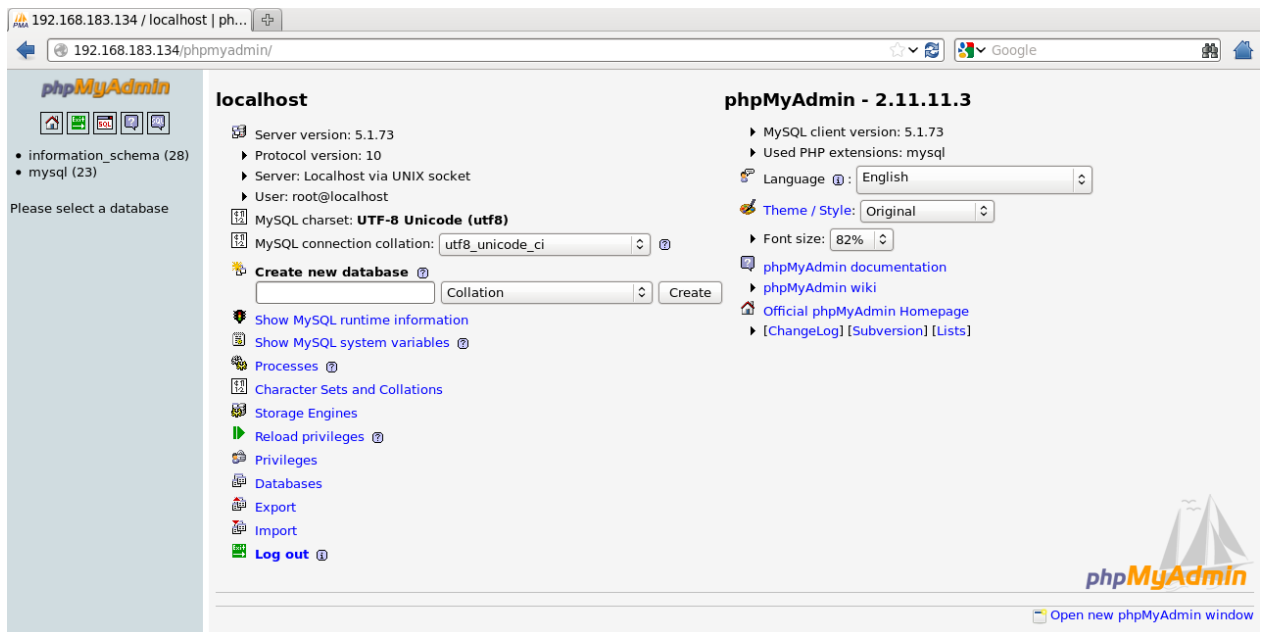


Рисунок 3.6 – Окно установленного phpMyAdmin

3.2 Проектирование БД

3.2.1 Модель данных

Проанализировав предметную область, мне удалось смоделировать модель базы данных состоящую из 27 таблиц, связанных между собой, которые содержат в себе все необходимые данные для описания компьютера/устройства.

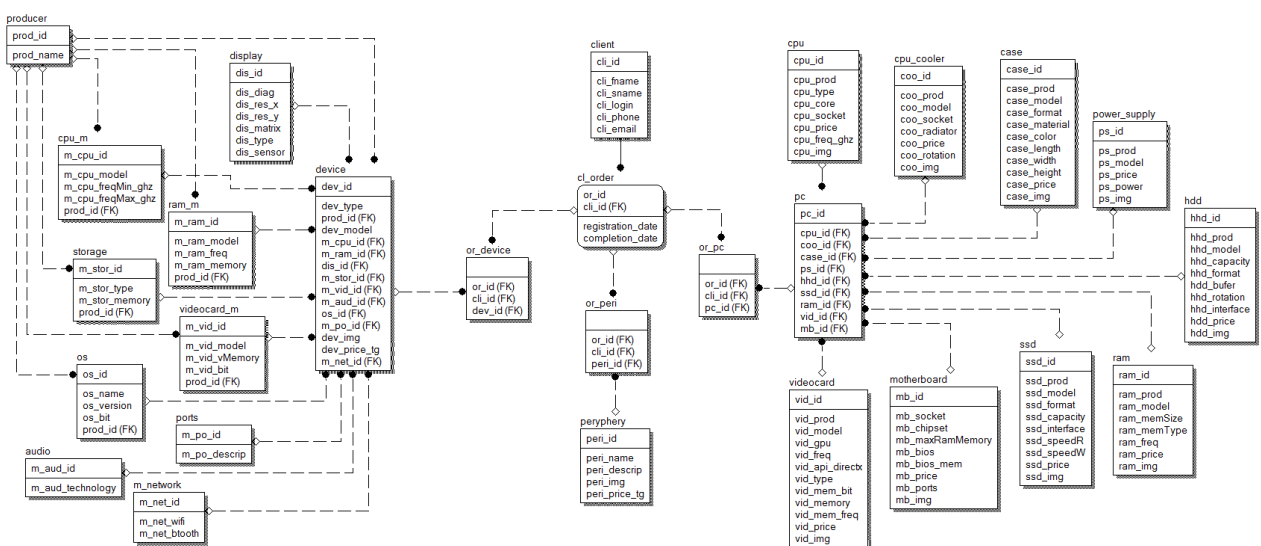


Рисунок 3.7 – Общая ER-диаграмма БД

3.2.2 Назначение сущностей предметной области

В таблице 1 описываются все 27 сущностей, присутствующей на модели базы данных.

Таблица 1 – Перечень сущностей

Наименование сущности	Назначение сущности
client	Информация о клиентах системы.
cl_order	Информация о заказах клиента.
or_pc	Конкретный заказ на компьютеры.
pc	Информация о собранном из комплектующих компьютере.
cpu	Информация о процессоре на компьютер.
cpu_cooler	Информация о кулере для процессора на компьютер.
pc_case	Информация о корпусе на компьютере.
power_supply	Информация о блоке питания на компьютер.
hdd	Информация о жестком диске на компьютер.
ram	Информация об оперативной памяти на компьютер.
ssd	Информация о твердотельном накопителе на компьютер.
motherboard	Информация о материнской плате на компьютер.
videocard	Информация о видео карте на компьютер.
or_device	Конкретный заказ на устройство.
device	Информация об устройстве.
producer	Информация о производителях.
display	Информация о дисплее устройства.
cpu_m	Информация о процессоре на устройстве.
ram_m	Информация об оперативной памяти на устройстве.
storage	Информация о памяти на устройстве.
videocard_m	Информация о видеокарте на устройстве.
os	Информация об операционной системе на устройстве.
ports	Информация о портах на устройстве.
audio	Информация об аудио технологий на устройстве.
network_m	Информация сетевом адаптере на устройстве.
or_peri	Конкретный заказ на периферию.
periphery	Информация о периферии.

3.2.3 Описание атрибутов сущностей

Таблица 2 – Определение атрибутов сущности «client»

Имя поля	Тип	Длина	Описание
cli_id	Числовой	11	Идентификатор клиента
cli_fname	Символьный	40	Фамилия клиента
cli_sname	Символьный	40	Имя клиента
cli_login	Символьный	40	Логин клиента
cli_phone	Числовой	11	Телефон клиента
cli_email	Символьный	40	Электронный почтовый ящик клиента

Таблица 3 – Определение атрибутов сущности «cl_order»

Имя поля	Тип	Длина	Описание
or_id	Числовой	11	Идентификатор заказа
cli_id	Числовой	11	Идентификатор клиента
registration_date	Дата	–	Дата регистрации заказа
completion_date	Дата	–	Дата выполнения заказа

Таблица 4 – Определение атрибутов сущности «or_pc»

Имя поля	Тип	Длина	Описание
or_id	Числовой	11	Идентификатор заказа
cli_id	Числовой	11	Идентификатор клиента
pc_id	Числовой	11	Идентификатор компьютера

Таблица 5 – Определение атрибутов сущности «pc»

Имя поля	Тип	Длина	Описание
pc_id	Числовой	11	Идентификатор компьютера
cpu_id	Числовой	11	Идентификатор процессора
coo_id	Числовой	11	Идентификатор кулера
case_id	Числовой	11	Идентификатор корпуса
ps_id	Числовой	11	Идентификатор блока питания
hdd_id	Числовой	11	Идентификатор жесткого диска
ssd_id	Числовой	11	Идентификатор твердотельного накопителя
ram_id	Числовой	11	Идентификатор оперативной памяти
vid_id	Числовой	11	Идентификатор видеокарты
mb_id	Числовой	11	Идентификатор материнской платы

Таблица 6 – Определение атрибутов сущности «cpu»

Имя поля	Тип	Длина	Описание
cpu_id	Числовой	11	Идентификатор процессора
cpu_prod	Символьный	40	Производитель процессора
cpu_type	Символьный	40	Модель процессора
cpu_core	Символьный	40	Кодовое название микроархитектуры процессора
cpu_socket	Символьный	40	Сокет процессора
cpu_freq_ghz	Числовой	40	Частота процессора
cpu_price_kz	Числовой	11	Цена процессора

Таблица 7 – Определение атрибутов сущности «cpu_cooler»

Имя поля	Тип	Длина	Описание
coo_id	Числовой	11	Идентификатор кулера для процессора
coo_prod	Символьный	40	Производитель кулера
coo_model	Символьный	40	Модель кулера
coo_socket	Символьный	40	Сокет кулера
coo_rotation_rpm	Числовой	11	Частота вращения кулера
coo_radiator	Символьный	40	Материал радиатора кулера
coo_price_tg	Числовой	11	Цена кулера

Таблица 8 – Определение атрибутов сущности «hdd»

Имя поля	Тип	Длина	Описание
hdd_id	Числовой	11	Идентификатор жесткого диска
hdd_prod	Символьный	40	Производитель жесткого диска
hdd_model	Символьный	40	Модель жесткого диска
hdd_capacity_mb	Числовой	11	Емкость жесткого диска
hdd_bufer_mb	Числовой	11	Буфер жесткого диска
hdd_rotation_rpm	Числовой	11	Частота вращения жесткого диска
hdd_format	Числовой	11	Физический размер жесткого диска
hdd_interface	Символьный	40	Интерфейс жесткого диска
hdd_price_tg	Числовой	11	Цена жесткого диска

Таблица 9 – Определение атрибутов сущности «pc_case»

Имя поля	Тип	Длина	Описание
case_id	Числовой	11	Идентификатор корпуса
case_prod	Символьный	40	Производитель корпуса
case_model	Символьный	40	Модель корпуса

case_format	Символьный	40	Тип размера корпуса
case_material	Символьный	40	Материал корпуса
case_color	Символьный	40	Цвет корпуса
case_length_cm	Числовой	11	Длина корпуса
case_height_cm	Числовой	11	Ширина корпуса
case_width_cm	Числовой	11	Высота корпуса
case_img	Символьный	255	Путь на изображение корпуса
case_price_tg	Числовой	11	Цена корпуса

Таблица 10 – Определение атрибутов сущности «power_supply»

Имя поля	Тип	Длина	Описание
ps_id	Числовой	11	Идентификатор блока питания
ps_prod	Символьный	40	Производитель блока питания
ps_model	Символьный	40	Модель блока питания
ps_power_w	Числовой	11	Мощность блока питания
ps_price_tg	Числовой	11	Цена блока питания

Таблица 11 – Определение атрибутов сущности «ram»

Имя поля	Тип	Длина	Описание
ram_id	Числовой	11	Идентификатор оперативной памяти
ram_prod	Символьный	40	Производитель оперативной памяти
ram_model	Символьный	40	Модель оперативной памяти
ram_memSize_gb	Числовой	11	Емкость оперативной памяти
ram_memType	Символьный	40	Тип оперативной памяти
ram_freq_mhz	Числовой	11	Частота оперативной памяти
ram_price_tg	Числовой	11	Цена оперативной памяти

Таблица 12 – Определение атрибутов сущности «ssd»

Имя поля	Тип	Длина	Описание
ssd_id	Числовой	11	Идентификатор твердотельного накопителя
ssd_prod	Символьный	40	Производитель твердотельного накопителя
ssd_model	Символьный	40	Модель твердотельного накопителя
ssd_capacity_gb	Числовой	11	Емкость твердотельного накопителя

ssd_format	Числовой	11	Физический размер твердотельного накопителя
ssd_interface	Символьный	40	Интерфейс твердотельного накопителя
ssd_speedR_mbs	Числовой	11	Скорость чтения твердотельного накопителя
ssd_speedW_mbs	Числовой	11	Скорость записи твердотельного накопителя
ssd_price_tg	Числовой	11	Цена твердотельного накопителя

Таблица 13 – Определение атрибутов сущности «motherboard»

Имя поля	Тип	Длина	Описание
mb_id	Числовой	11	Идентификатор материнской платы
mb_type	Символьный	40	Тип материнской платы
mb_socket	Символьный	40	Сокет материнской платы
mb_chipset	Символьный	40	Чипсет материнской платы
mb_maxRamMemory_gb	Числовой	11	Максимальная емкость оперативной памяти
mb_bios	Символьный	40	Микропрограмма материнской платы
mb_ports	Символьный	40	Порты материнской платы
mb_img	Символьный	255	Путь на изображение материнской платы
mb_price_tg	Числовой	11	Цена материнской платы

Таблица 14 – Определение атрибутов сущности «videocard»

Имя поля	Тип	Длина	Описание
vid_id	Числовой	11	Идентификатор видеокарты
vid_prod	Символьный	40	Производитель видеокарты
vid_model	Символьный	40	Модель видеокарты
vid_gpu	Символьный	40	Процессор видеокарты
vid_freq_mhz	Числовой	11	Частота процессора видеокарты
vid_api	Символьный	40	Поддержка API видеокарты
vid_mem_byte	Числовой	11	Разрядность шины видеопамати
vid_memory_gb	Числовой	11	Видеопамать видеокарты
vid_type	Символьный	40	Тип видеопамати видеокарты
vid_mem_freq_mhz	Числовой	11	Частота видеокарты
vid_price_tg	Числовой	11	Цена видеокарты

Таблица 15 – Определение атрибутов сущности «or_peri»

Имя поля	Тип	Длина	Описание
or_id	Числовой	11	Идентификатор заказа
cli_id	Числовой	11	Идентификатор клиента
peri_id	Числовой	11	Идентификатор периферии

Таблица 16 – Определение атрибутов сущности «periphery»

Имя поля	Тип	Длина	Описание
peri_id	Числовой	11	Идентификатор периферии
prod_id	Числовой	11	Идентификатор производителя
peri_name	Символьный	40	Наименование периферии
peri_descrip	Символьный	40	Описание периферии
peri_img	Символьный	255	Путь на изображение периферии
peri_price_tg	Числовой	11	Цена периферии

Таблица 17 – Определение атрибутов сущности «or_device»

Имя поля	Тип	Длина	Описание
or_id	Числовой	11	Идентификатор заказа
cli_id	Числовой	11	Идентификатор клиента
dev_id	Числовой	11	Идентификатор устройства

Таблица 18 – Определение атрибутов сущности «device»

Имя поля	Тип	Длина	Описание
dev_id	Числовой	11	Идентификатор устройства
dev_type	Символьный	40	Тип устройства
dev_prod	Символьный	40	Производитель устройства
dev_model	Символьный	40	Модель устройства
m_cpu_id	Числовой	11	Идентификатор процессора
m_ram_id	Числовой	11	Идентификатор оперативной памяти
dis_id	Числовой	11	Идентификатор дисплея
m_stor_id	Числовой	11	Идентификатор физической памяти
m_vid_id	Числовой	11	Идентификатор видеокарты
m_aud_id	Числовой	11	Идентификатор аудио технологии
os_id	Числовой	11	Идентификатор операционной системы
m_po_id	Числовой	11	Идентификатор портов

m_net_id	Числовой	11	Идентификатор сетевого адаптера
dev_img	Символьный	40	Путь на изображение устройства
dev_price_tg	Числовой	11	Цена устройства

Таблица 19 – Определение атрибутов сущности «producer»

Имя поля	Тип	Длина	Описание
prod_id	Числовой	11	Идентификатор производителя
prod_name	Символьный	40	Наименование производителя

Таблица 20 – Определение атрибутов сущности «display»

Имя поля	Тип	Длина	Описание
dis_id	Числовой	11	Идентификатор дисплея
dis_diag_inch	Числовой	11	Диагональ дисплея
dis_res_x	Числовой	11	Разрешение по оси x дисплея
dis_res_y	Числовой	11	Разрешение по оси y дисплея
dis_matrix	Символьный	40	Матрица дисплея
dis_type	Символьный	40	Тип дисплея
dis_sensor	Числовой	11	Наличие сенсорного дисплея

Таблица 21 – Определение атрибутов сущности «cpu_m»

Имя поля	Тип	Длина	Описание
m_cpu_id	Числовой	11	Идентификатор процессора
prod_id	Числовой	11	Идентификатор производителя
m_cpu_model	Символьный	40	Модель процессора
m_cpu_freqMin_ghz	Числовой	11	Минимальная частота процессора
m_cpu_freqMax_ghz	Числовой	11	Максимальная частота процессора

Таблица 22 – Определение атрибутов сущности «ram_m»

Имя поля	Тип	Длина	Описание
m_ram_id	Числовой	11	Идентификатор оперативной памяти
m_ram_model	Символьный	40	Модель оперативной памяти
m_ram_freq_ghz	Числовой	11	Частота оперативной памяти
m_ram_memory_gb	Числовой	11	Емкость оперативной памяти

Таблица 23 – Определение атрибутов сущности «storage»

Имя поля	Тип	Длина	Описание
m_stor_id	Числовой	11	Идентификатор накопителя
m_stor_type	Символьный	40	Тип накопителя
m_stor_memory_gb	Числовой	11	Емкость накопителя

Таблица 24 – Определение атрибутов сущности «videocard_m»

Имя поля	Тип	Длина	Описание
m_vid_id	Числовой	11	Идентификатор видеокарты
prod_id	Числовой	11	Идентификатор производителя
m_vid_model	Символьный	40	Модель видеокарты
m_vid_memory_gb	Числовой	11	Емкость памяти видеокарты
m_vid_byte	Числовой	11	Разрядность шины видеопамати

Таблица 25 – Определение атрибутов сущности «os»

Имя поля	Тип	Длина	Описание
os_id	Числовой	11	Идентификатор операционной системы
prod_id	Числовой	11	Идентификатор производителя
os_name	Символьный	40	Наименование операционной системы
os_bit	Числовой	11	Архитектура операционной системы

Таблица 26 – Определение атрибутов сущности «ports»

Имя поля	Тип	Длина	Описание
m_po_id	Числовой	11	Идентификатор портов
m_po_descrip	Символьный	–	Описание портов

Таблица 27 – Определение атрибутов сущности «audio»

Имя поля	Тип	Длина	Описание
m_aud_id	Числовой	11	Идентификатор аудио технологии
m_aud_technology	Символьный	40	Описание аудио технологии

Таблица 28 – Определение атрибутов сущности «network_m»

Имя поля	Тип	Длина	Описание
m_net_id	Числовой	11	Идентификатор сетевого адаптера
m_net_btooth	Числовой	11	Версия Bluetooth
m_net_wifi	Символьный	40	Поддерживаемые стандарты

Данные таблицы определяют атрибуты каждой таблицы в модели базы данных. Все они являются необходимыми для вывода информации об устройстве.

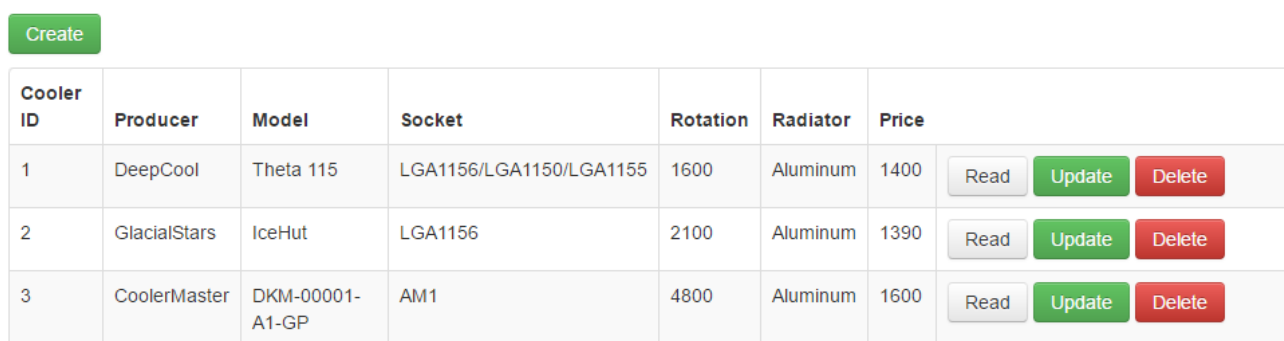
3.3 Разработка пользовательского интерфейса

Разработка пользовательского интерфейса всегда была главной задачей программиста, так как он играет связующую роль между пользователем и разработанной системой. Так же программист-разработчик должен найти правильный подход при создании пользовательского интерфейса, потому что существуют несколько видов факторов, влияющих на пользователя:

- Функциональность;
- Производительность;
- Эстетика.

Разработанный мною пользовательский интерфейс написан на PHP, который напрямую связан с локальным сервером. Интерфейс связан с базой данных, а точнее с таблицами, пользователь интуитивно разберется в интерфейсе, так как он очень простой.

PHP CRUD Grid



The screenshot shows a web interface for a PHP CRUD Grid. At the top left, there is a green 'Create' button. Below it is a table with the following columns: Cooler ID, Producer, Model, Socket, Rotation, Radiator, Price, and a set of action buttons (Read, Update, Delete). The table contains three rows of data.

Cooler ID	Producer	Model	Socket	Rotation	Radiator	Price	Read	Update	Delete
1	DeepCool	Theta 115	LGA1156/LGA1150/LGA1155	1600	Aluminum	1400	Read	Update	Delete
2	GlacialStars	IceHut	LGA1156	2100	Aluminum	1390	Read	Update	Delete
3	CoolerMaster	DKM-00001-A1-GP	AM1	4800	Aluminum	1600	Read	Update	Delete

Рисунок 3.8 – Пользовательский интерфейс базы данных

4 Техничко-экономическое обоснование

4.1 Описание работы и обоснование необходимости

Тема данного дипломного проекта – «Разработка базы данных для мобильного решения технического обслуживания и ремонта». Данный проект ориентирован на хранение данных о товаре в базе данных для мобильного приложения, а также на уменьшение затрат времени и функциональности. Цель работы - создание базы данных, которая ускорит работу мобильного приложения, выполняющее необходимые функции решения технического обслуживания и ремонта. [13]

4.2 Расчет затрат на разработку информационных технологий

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

$$C_{ni} = Z_{\text{фот}} + Z_{\text{сзи}} + M_i + P_{\text{си}} + P_{\text{ми}} + P_{\text{нки}} + \Pi_{\text{зи}} + P_{\text{ни}} \quad (4.1)$$

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

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

M_i – затраты на материалы, тенге;

$P_{\text{си}}$ – затраты на специальные программные средства, необходимые для разработки проектного решения, тенге;

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

$P_{\text{нки}}$ – затраты на научные командировки, тенге;

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

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

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

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

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

Z_{di} – дополнительная заработная плата, тенге.

Общая трудоемкость проекта рассчитывается по формуле:

$$T_o = T_n * K_c * K_t * K_n \quad (4.3)$$

где K_c – коэффициент, учитывающий сложность ПО;

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

K_n – коэффициент, учитывающий степень новизны ПО;

T_n – нормативная трудоемкость.

Расчет коэффициента сложности производится на основе данных, представленных в таблице 1 и составляет $K_c = 0,26$ т.к. в ПО присутствует более 3-х характеристик – обеспечение хранения, ведения и поиска данных в сложных структурах, интерактивный доступ, функционирование программного обеспечения в расширенной операционной среде.

Таблица 29 – Дополнительные коэффициенты сложности ПО

Характеристика ПО	Значения K_c
1. Функционирование ПО в расширенной операционной среде (связь с другими ПО)	0,08
2. Интерактивный доступ	0,06
3. Обеспечение хранения, ведения и поиска данных в сложных структурах	0,07
4. Наличие у ПО одновременно нескольких характеристик	
– 2 характеристики	0,12
– 3 характеристики	0,18
– Свыше 3-х характеристик	0,26

Поправочный коэффициент, учитывающий степень использования при разработке проекта стандартных модулей (K_T), определяется на основе данных представленных в таблице 2 и составляет 0,8.

Таблица 30 – Значения поправочного коэффициента, учитывающего использование стандартных модулей типовых программ и ПО (K_T)

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

Поправочный коэффициент, учитывающий новизну разрабатываемого проекта (K_n) определяется на основе данных, представленных в таблице 3 и составляет 0,7.

Таблица 31 – Поправочные коэффициенты, учитывающие новизну ПО (K_n)

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

Базой для определения нормативной трудоемкости являются, укрупненные нормы времени на разработку проекта в зависимости от уточненного объема ПО и группы сложности. Нормативная трудоемкость проекта (T_n) определяется на основе принятого в расчет объема ПП и категории сложности, которая уточняется с учетом сложности и новизны проекта и степени использования стандартных модулей при разработке. Учитывая данные из Приложения А: для 1-ой категории сложности ПО

$$T_n = 520$$

Рассчитаем общий объем трудоемкости исходя из формулы

$$T_o = 520 \cdot 0,26 \cdot 0,8 \cdot 0,7 = 76 \text{ чел./дн.}$$

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

– расчет числа исполнителей при заданных сроках разработки проекта;

– определение сроков разработки проекта при заданной численности исполнителей.

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

$$Ч = T_0 / (T_p * \Phi_{эф}) \quad (4.4)$$

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

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

T_p – срок разработки проекта (лет).

Срок разработки проекта (T_p) определяется по формуле

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

где $Ч_p$ – плановое число разработчиков.

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

$$\Phi_{эф} = D_r - D_{п} - D_{в} - D_o \quad (4.6)$$

где D_r – количество дней в году;

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

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

D_o – количество дней отпуска.

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

$$D_r = 366;$$

$$D_{п} = 16;$$

$$D_{в} = 103;$$

$D_o = 14$, эффективный фонд времени одного работника составит

$$\Phi_{эф} = 366 - 16 - 103 - 14 = 233 \text{ дня}$$

Плановое число разработчиков $Ч_p = 1$, следовательно, по формуле

$$T_p = 76 / (1 * 233) = 0,33 = 121 \text{ дней}$$

Таким образом, согласно произведенным расчетам и в соответствие с формулой:

$$Ч = \frac{76}{0,33 \cdot 233} = 1 \text{ человек}$$

Основная заработная плата исполнителей на конкретное ПО рассчитывается по формуле:

$$З_{oi} = \sum_{i=1}^n T_{чи} \cdot T_{ч} \cdot K \quad (4.7)$$

где n – количество исполнителей, занятых разработкой конкретного ПО;

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

$\Phi_{п}$ – плановый фонд рабочего времени i -го исполнителя (дней), 121 раб день;

$T_{ч}$ – количество часов работы в день (час), 8 часов;

K – коэффициент премирования, составляет 1,38.

По данным о специфике и сложности выполняемых функций составляется штатное расписание группы специалистов–исполнителей, участвующих в разработке ПО, с определением образования, специальности, квалификации и должности (таблица 32).

Таблица 32 – Сведения по работникам, задействованным в проекте

Специалист–Исполнитель	Количество, человек	Заработная плата в месяц, тенге
Инженер–программист	1	80000
Итого	1	80000

Часовая тарифная ставка рассчитывается путем деления месячной тарифной ставки, установленную при 48–часовой недельной норме рабочего времени и общего фонда времени (Φ_p):

$$T_{ч} = T_{м} / \Phi_p \quad (4.8)$$

где $T_{ч}$ – часовая тарифная ставка (тыс.тенге);

$T_{м}$ – месячная тарифная ставка (тыс.тенге).

Общий фонд времени:

$$\Phi_p = T_{ч} * \Phi_{п} \quad (4.9)$$

Таким образом:

$$\Phi_p = 8 * 26 = 208 \text{ час.}$$

Рассчитаем тарифную ставку инженера–программиста:

$$T_{\text{ч}} = \frac{80\,000}{208} = 385 \text{ тенге/час}$$

В соответствие с формулой основная заработная плата инженера–программиста составит:

$$З_{oi} = 385 * 8 * 26 * 1,38 = 110510 \text{ тенге}$$

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

$$З_{czi} = (\text{ФОТ} - \text{ПО}) * 11\% \quad (4.10)$$

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

$$\text{ПО} = \text{ФОТ} * 10\% \quad (4.11)$$

Таким образом:

$$\text{ПО} = 110510 * 0,1 = 11051 \text{ тенге,}$$

$$З_{czi} = (110510 - 11051) * 0,11 = 10940,5 \text{ тенге}$$

Затраты на материалы определяются по формуле

$$M_i = (З_{\text{осн}} * N_{\text{мз}}) / 100\% \quad (4.12)$$

где $N_{\text{мз}}$ – норма расхода материалов основной заработной платы (3–5%).

$$M_i = 110510 * 0,05 = 5525,5 \text{ тенге}$$

Расходы по статье «Спецоборудование» (P_{ci}) включают затраты средств на приобретение вспомогательных специального назначения технических и

программных средств, необходимых для разработки конкретного ПО включая расходы на их проектирование, изготовление, отладку, установку и эксплуатацию:

$$P_{ci} = \sum_{i=1}^n C_{ci} \quad (4.13)$$

Таблица 33 – Перечень оборудования, необходимого для разработки

Наименование изделий	Характеристика	Количество единиц	Цена за единицу, тенге	Общая сумма, тенге
Ноутбук	Lenovo Yoga 3 Pro	1	400000	400000
Итого				400000

Таблица 34 – Программное обеспечение, используемое в работе

Программное обеспечение	Стоимость, тенге
Rational Rose	170000
XAMPP Control Panel	бесплатно
ERwin Data Modeler r7	бесплатно
Итого	170000

Амортизационные отчисления рассчитываются по формуле:

$$A_i = \frac{N_A * C_{ПЕР} * N}{100 * n} \quad (4.14)$$

где N_A – норма амортизации;

$C_{ПЕР}$ – первоначальная стоимость оборудования;

N – количество дней на выполнение работ;

n – количество рабочих дней в году.

Таким образом амортизационные отчисления по используемому оборудованию составляют:

$$A_{об} = \frac{400000 * 25 * 121}{100 * 366} = 33060 \text{ тенге}$$

Амортизационные отчисления по используемому программному обеспечению составляют:

$$A_{пр} = \frac{170000 * 25 * 121}{100 * 366} = 14050 \text{ тенге,}$$

$$A_i = 33060 + 14050 = 47110 \text{ тенге}$$

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

$$P_{zi} = Z_{oi} * N_{пз}/100 \quad (4.15)$$

где $N_{пз}$ – норматив прочих затрат в целом по организации (20%).
Таким образом:

$$P_{zi} = 514298 \cdot 0,2 = 102859,6 \text{ тенге}$$

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

$$P_{ni} = Z_{oi} * N_{рн}/100\% \quad (4.16)$$

где P_{ni} – накладные расходы на конкретную ПО (тыс.тенге);
 $N_{рн}$ – норматив накладных расходов в целом по организации (70%).
Таким образом накладные расходы составят

$$P_{ni} = 514298 \cdot 0,7 = 360008,6 \text{ тенге}$$

Полные затраты на разработку базы данных для мобильного приложения продажи компьютеров составят:

$$C_{ni} = 514298 + 50915,5 + 25714,9 + 47110 + 102859,6 + 360008,6 = 1100906,6 \text{ тенге.}$$

Сводные результаты расчета затрат на разработку ПО и их структура представлены в таблице 35 и на рисунке 4.1.

Таблица 35 – Затраты на разработку базы данных для мобильного приложения продажи компьютеров

Затраты на разработку	Условное обозначение	Значение, тенге	В процентах от общей суммы
-----------------------	----------------------	-----------------	----------------------------

Фонд оплаты труда	$Z_{\text{ФОТ}}$	514298	46,7
Социальный налог	$Z_{\text{сзи}}$	50915,5	4,6
Материалы	M_i	25714,9	2,3
Амортизационные отчисления	$A_{\text{об}}$	47110	4,3
Прочие затраты	$P_{\text{зи}}$	102859,6	9,3
Накладные расходы	$P_{\text{ни}}$	360008,6	32,7
Итого:		1100906,6	100



Рисунок 4.1 – Структура затрат на разработку ПП

4.3 Расчет цены программного продукта

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

$$C_{\text{ПП}} = Z_{\text{РПР}} + P_{\text{П}} + \text{НДС} \quad (4.17)$$

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

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

$P_{\text{П}}$ – планируемая прибыль, тенге;

НДС – налог на добавленную стоимость, тенге.

Планируемая прибыль составляет (20%) от себестоимости разработки

$$П_{п} = З_{рпр} \cdot 0,2 \quad (4.18)$$

$$П_{п} = 1100906,6 \cdot 0,2 = 220181,3 \text{тенге}$$

НДС, начисленный на $П_{п}$, определяется следующим образом:

$$\text{НДС} = (П_{п} + З_{рпр}) \cdot k_{\text{НДС}} \quad (4.19)$$

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

Подставив данные в формулу получаем

$$\text{НДС} = (1100906,6 + 220181,3) \cdot 0,12 = 158530,5 \text{ тенге}$$

Цена реализации $П_{п}$ по формуле составляет

$$Ц_{пп} = 1100906,6 + 220181,3 + 158530,5 = 1479618,4 \text{ тенге}$$

4.4 Оценка социально-экономических результатов

Данный ПП будет востребован на рынке, так как имеет интуитивно понятный интерфейс, а его цена близка к обще рыночной.

Кроме того: уменьшается время на поиски и добавления новых товаров, облегчается труд сотрудника, все клиенты будут довольны быстрым обслуживанием, автоматически будет подсчитываться заказ на определённую сумму в зависимости от наименования и количества товаров.

5 Безопасность жизнедеятельности

5.1 Анализ условий труда

Разработка БД для мобильного приложения магазина электронной техники осуществляется с использованием компьютерной техники и электронного оборудования. В рассматриваемом помещении работают разработчик, дизайнер и администратор, которые имеют свое рабочее место.

При разработке программного продукта важную роль играет правильная организация условий труда в рабочем помещении.

Так как данная работа связана с длительным нахождением за компьютером, необходимо учесть нормы искусственного освещения.

Анализ пожарной безопасности и кондиционирования для обеспечения благоприятных условий работы.

Рассмотрим пример типового производственного помещения. В комнате, помимо оборудования и рабочих мест, могут находиться различные обогреватели и кондиционеры. Помещение, в котором были установлены три компьютера. В комплекс вычислительной аппаратуры дополнительно входят: графопостроитель, лазерный принтер, матричный принтер.

Оператор программист как пользователь программы на своем рабочем месте подвергается следующим вредным факторам:

- неблагоприятные микроклиматические условия, которые определяются температурой воздуха, его составом и давлением, относительной влажностью, скоростью движения воздушных потоков;
- неправильная освещенность рабочего места, вследствие неоптимального расположения и использования осветительных приборов;
- эксплуатационные шумы ПЭВМ, вентиляционных устройств, кондиционеров и внешние шумы;
- электромагнитное излучение, главным источником которого является монитор;
- психофизиологические факторы, т.е. умственные перенапряжение зрительных и слуховых анализаторов, монотонность труда, эмоциональные перегрузки и другие.

Кроме этого оператор подвергается таким опасностям, как:

- опасность поражения электрическим током $U=220$ В, $f=50$ Гц; при пользовании электроприборами, в частности компьютером;
- опасность возникновения пожара вследствие ошибочного подключения электроприборов.

Длительное нахождение человека в зоне комбинированного воздействия различных неблагоприятных факторов может привести к профессиональному

заболеванию. Следует отметить, что персонал компьютерной лаборатории работает в условиях постоянной опасности.

5.2 План помещения

План рабочего помещения представлен на рисунке 5.1.

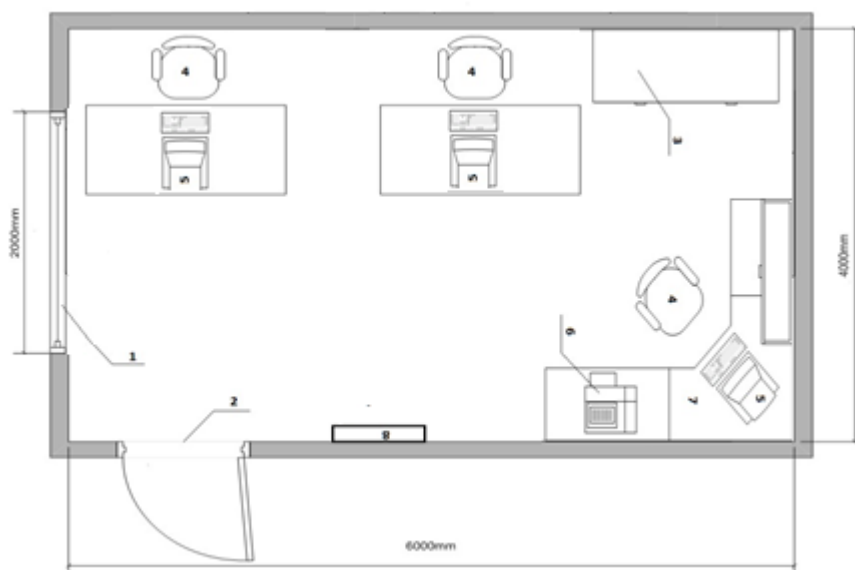


Рисунок 5.1 – Рабочее помещение

Где 1 - шкаф,
2 - стулья,
3 - персональные,
4 - компьютеры,
5 - принтер,
6 - столы,
7 - кондиционер,
8 - окна,
9 - дверь.

- Помещение имеет следующие параметры:
- размеры рабочей аудитории:
 - 1) высота - 3,2 м;
 - 2) ширина - 4 м;
 - 3) длина - 6 м.
- общая площадь помещения составляет 24 м²;
- площадь, занимаемая оборудованием и мебелью ≈ 15 м²;
- помещение по зрительным условиям работы относится к категории легких работ (легкая физическая, категория I_a , работа производится сидя и не требует физического напряжения) ГОСТ 12.2.032-78 «ССБТ Общие эргономические требования»;

– искусственное освещение - светильники: 2 светильника, в каждом по 2 люминесцентные лампы.

5.3 Характеристики используемого оборудования

При разработке и дальнейшей эксплуатации программного продукта используется следующее оборудование:

- Персональные компьютеры - 3 шт;
- Люминесцентные лампы Т8\58W, 2 светильника по 2 лампы E=400лк;
- Источник бесперебойного питания: переменное напряжение 220-250 В, частотой 50 Гц. Мощность 400 Вт;
- Кондиционер SAMSUNG Monte.

В таблице приведены оптимальные нормы параметров микроклимата с учетом периода года согласно ГОСТ 12.0.003-88. «ССБТ для легкой физической работы». Оборудование, установленное в рабочем помещении, не является источником выделения тепла (очень незначительное выделение тепла аппаратурой никаким образом не оказывает влияние на микроклимат рабочего помещения).

Таблица 36 – Оптимальные нормы микроклимата для помещений с ПК

Период года	Категория работ	Температура воздуха °С не более	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	Легкая - 1а	22-24	40-60	0,1
Теплый	Легкая - 1а	23-25	40-60	0,1

Здание относится к I степени огнестойкости (СНиП РК 2.02-05-2002). Рабочее помещение по вопросам пожарной безопасности относится к классу “Д”. В соответствии с типовыми правилами пожарной безопасности административные здания и отдельные помещения, и технологические установки обеспечиваются первичными средствами пожаротушения согласно нормативам.

5.4 Расчет мер защиты от поражения электрическим током

Заземления – устранение опасности поражения током в случае прикосновения к корпусу электроустановки и другим нетоковедущим металлическим частям, оказавшимся под напряжением вследствие замыкания на корпус и по другим причинам. Как известно, контур заземления состоит из вертикальных заземлителей, горизонтальных заземлителей и заземляющего проводника. Вертикальные заземлители вбиваются в почву на определенную глубину. Горизонтальные заземлители соединяют между собой вертикальные заземлители. Заземляющий проводник соединяет контур заземления

непосредственно с электрощитом. Размеры и количество этих заземлителей, расстояние между ними, удельное сопротивление грунта – все эти параметры напрямую зависят от сопротивления заземления. Заземление служит для снижения напряжения прикосновения до безопасной величины. Благодаря заземлению опасный потенциал уходит в землю тем самым, защищая человека от поражения электрическим током. Величина тока стекания в землю зависит от сопротивления заземляющего контура. Чем сопротивление будет меньше, тем величина опасного потенциала на корпусе поврежденной электроустановки будет меньше. Заземляющие устройства удовлетворяют возложенным на них определенным требованиям, а именно величины сопротивления растекания токов и распределения опасного потенциала. Основным расчет защитного заземления сводится к определению сопротивления растекания тока заземлителя. Это сопротивление зависит от размеров и количества заземляющих проводников, расстояния между ними, глубины их заложения и проводимости грунта.

Работу необходимо проводить в соответствии с правилами технической эксплуатации электроустановок. Классификация помещений по опасности поражения электрическим током, по квалификации персонала и напряжению сетей. Также необходимы вводный и периодический инструктажи по технике безопасности при работе с электрическими приборами, соблюдать трудовую дисциплину, правильно организовать рабочее место.

Шины защитного заземления расположены в местах, доступных для проверки и сохранности. Технические меры защиты. Для защиты от прикосновения к токоведущим частям оборудования и приборов используют изоляцию, недоступное расположение токоведущих частей и ограждение. Для защиты от поражения электрическим током при прикосновении к металлическим частям оборудования, которые могли случайно оказаться под напряжением, выполнено защитное заземление корпуса установки, $R < 4 \text{ Ом}$.

Сопротивление одиночного заземлителя определяется по формуле

$$R_{\text{оз}} = 0,366 \cdot \rho \cdot \lg \frac{2L}{d} + 0,5 \cdot \frac{\lg \frac{2 \cdot 4 \cdot t + L}{4t \cdot L}}{L} \quad [17] \quad (5.1)$$

Где, $R_{\text{оз}}$ – сопротивление одиночного заземлителя, Ом;

ρ – удельное сопротивление грунта, Ом.м;

L – длина заземлителя, м;

t – глубина заложения заземлителя, м;

d – диаметр заземлителя, м.

В используемой установке применяются заземлители из угловой стали,

для которых определяется эквивалентный диаметр

$$d_{\text{ЭКВ}} = 0,95 \cdot b \text{ [16]} \quad (5.2)$$

где b – ширина сторон уголка, м.

$$d_{\text{ЭКВ}} = 0,95 \cdot 0,05 = 0,0475$$

Формула (5.1) для заземлителя из угловой стали примет вид:

$$R_{\text{ОЗ}} = 0,366 \cdot p \cdot \lg \frac{2L}{d_{\text{ЭКВ}}} + 0,5 \cdot \frac{\lg \frac{2 \cdot 4 \cdot t + L}{4t \cdot L}}{L}, \quad (5.3)$$

$$R_{\text{ОЗ}} = 0,366 \cdot 50 \cdot \lg \frac{2 \cdot 2,5}{0,0475} + 0,5 \cdot \frac{\lg \frac{2 \cdot 4 \cdot 70 + 2,5}{-2,5}}{2,5}$$

Из формулы 5.3 следует:

$$R_{\text{ОЗ}} = 17,5 \text{ Ом.}$$

Количество заземлителей определяется по формуле:

$$n = R_{\text{ОЗ}} / R_{\text{ЗН}} \text{ [16]} \quad (5.4)$$

где n – число заземлителей, шт;

$R_{\text{ОЗ}}$ – сопротивление одиночного заземлителя, Ом;

$R_{\text{ЗН}}$ – сопротивление заземлителя по нормам (4 Ом).

Расстояние между заземлителями рассчитывается по формуле:

$$a = 2 \cdot L \text{ [17]} \quad (5.5)$$

где a – расстояние между заземлителями, м;

L – длина заземлителя, м.

$$a = 2 \cdot 2,5 = 5 \text{ м.}$$

Число заземлителей с учетом их взаимного экранирования определяется по формуле

$$n_{\text{э}} = n / \eta_{\text{эз}} \quad [18] \quad (5.6)$$

где, $n_{\text{э}}$ – число заземлителей с учетом их взаимного экранирования, шт.;
 n – число заземлителей без учета их взаимного экранирования, шт.;
 $\eta_{\text{эз}}$ – коэффициент использования заземлителей учитывающий их взаимное экранирование.

$$n_{\text{э}} = 4 / 0,88 = 5 \text{ шт.}$$

Длина заземляющих проводников определяется по формуле

$$L_n = 1,05 \cdot a \cdot n_{\text{э}} \quad [17] \quad (5.7)$$

где L_n – длина заземляющих проводников, м;
 a – расстояние между заземлителями;
 $n_{\text{э}}$ – число заземлителей с учетом их взаимного экранирования.

$$L_n = 1,05 \cdot 5 \cdot 5; \quad L_n = 26,25 \text{ м}$$

Сопротивление заземляющего проводника находится по формуле

$$R_{\text{ПП}} = 0,36 \cdot \rho \cdot (\lg(2 \cdot L_n / b \cdot t)) / L \quad [16] \quad (5.8)$$

где $R_{\text{ПП}}$ – сопротивление заземляющего проводника из полосовой стали, Ом;
 L_n – длина заземляющих проводников, м;
 b – ширина полосы заземляющего проводника, м;
 t – глубина заложения заземлителя, м

$$R_{\text{ПП}} = 0,366 \cdot 50 \cdot (\lg(2 \cdot 26,252 / 0,05 \cdot 0,7)) / 2,5,$$

$$R_{\text{ПП}} = 3,89 \text{ Ом}$$

Действительное сопротивление заземляющего проводника с учетом взаимного экранирования заземлителей и заземляющих проводников находится по формуле

$$R_{\text{ПП}} = R_{\text{ЗП}} / \eta_n \quad [16] \quad (5.9)$$

где $R_{\text{ПП}}$ – действительное сопротивление заземляющего проводника, Ом;

$R_{\text{ЗП}}$ – сопротивление заземляющего проводника, Ом;

η_n – коэффициент использования заземляющего проводника

$$R_{\text{ПП}} = \frac{3,89}{0,89} = 4,37 \text{ Ом}$$

Сопротивление растеканию тока всего заземляющего устройства находится по формуле

$$R_{\text{ЗП}} = R_{\text{ОЗ}} \cdot R_{\text{ПП}} / (R_{\text{ОЗ}} \cdot \eta_n + R_{\text{ПП}} \cdot \eta_{\text{эз}} \cdot n) \quad [17] \quad (5.10)$$

где $R_{\text{ЗП}}$ – сопротивление растеканию тока всего заземляющего устройства, Ом

$$R_{\text{ЗП}} = 17,5 \cdot 4,37 / (17,5 \cdot 0,89 + 4,37 \cdot 0,88 \cdot 4),$$

$$R_{\text{ЗП}} = 2,47 \text{ Ом}$$

Действительное число заземлителей определяется по формуле

$$n = R_{\text{ОЗ}} / \eta_{\text{эз}} \cdot R_{\text{ЗП}} \quad [18] \quad (5.11)$$

где n – действительное число заземлителей, шт.

$$n = 17,5 / (0,88 \cdot 2,47) = 8 \text{ шт.}$$

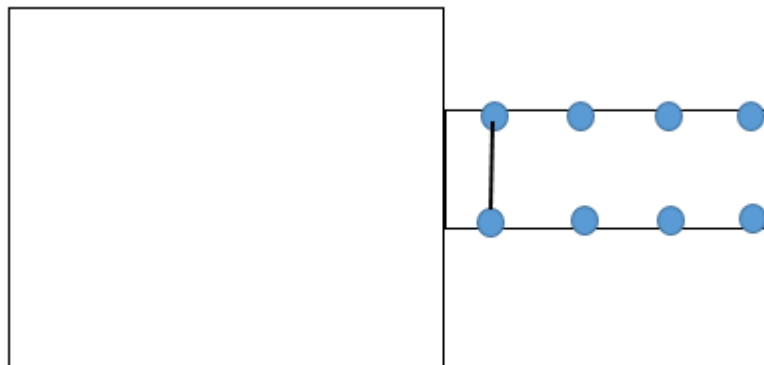


Рисунок 5.2 – Схема заземляющего устройства (ЗУ)

5.5 Расчет системы кондиционирования

В Таблице 5.2 приведены оптимальные нормы параметров микроклимата с учетом периода года согласно СНиП 2.04.05-91. для легкой физической работы.

Таблица 37 – Оптимальные нормы микроклимата для помещений с ПК

Период года	Категория работ	Температура воздуха °С не более	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	Легкая - 1а	22-24	40-60	0,1
Теплый	Легкая - 1а	23-25	40-60	0,1

Оборудование, установленное в рабочем помещении, не является источником выделения тепла (очень незначительное выделение тепла аппаратурой никаким образом не оказывает влияние на микроклимат рабочего помещения). Климатические условия эксплуатации оборудования полностью совпадают с климатическими условиями, нормируемыми для рабочего персонала.

Для вентиляции офисного помещения используются каналы естественной вентиляции, прокладываемые при строительстве здания и открытые окна летом. В теплый период года для поддержания оптимального микроклимата используется кондиционер. Нормальный микроклимат в офисе обеспечивает хорошее самочувствие сотрудников в любое время года, и соответственно продуктивность работы увеличивается. Таким образом, для поддержания условий микроклимата в помещении, целесообразно оборудовать его системой кондиционирования.

Ниже представлен расчет системы кондиционирования в рабочем помещении. Кондиционирование обеспечит соответствие климата в рабочем помещении нормативам.

Количество приточного воздуха $L_{пр}$, $\frac{м}{ч}$ определяем по формуле

$$L_{пр} = \frac{Q_{изб}}{c \cdot \rho_{пр} \cdot (t_{выт} - t_{пр})} \quad [15] \quad (5.12)$$

где $Q_{изб}$ – избыточное выделение явной теплоты, $\frac{кДж}{ч}$;

c – удельная теплоемкость воздуха при постоянном давлении, равная

$$c = 1 \frac{кДж}{кг^{\circ}C}$$

$\rho_{\text{пр}}$ – плотность поступающего в помещение воздуха, равная $1,2 \frac{\text{кг}}{\text{м}^3}$;

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

$t_{\text{пр}}$ – температура приточного воздуха, $^{\circ}\text{C}$.

Температура удаляемого из помещения воздуха $t_{\text{выт}}$, $^{\circ}\text{C}$, определяется по формуле:

$$t_{\text{выт}} = t_{\text{рз}} + \Delta t \cdot (h_{\text{вп}} - Z)^{[15]} \quad (5.13)$$

где $t_{\text{рз}}$ – температура в рабочей зоне, которая не должна превышать допустимую по нормам ($t_{\text{рз}} < t_{\text{доп}}$), $^{\circ}\text{C}$;

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

Z – высота рабочей зоны, м.

Поскольку расчет производится для теплого периода года, то примем:

$$t_{\text{рз}} = 22 \text{ } ^{\circ}\text{C}$$

Внутренняя часть кондиционера расположена на высоте $h_{\text{вп}} = 2,8\text{м}$:

$$t_{\text{выт}} = 22 + 1,2 \cdot (2,8 - 3,2) \text{ } ^{\circ}\text{C}$$

Температура приточного воздуха $t_{\text{пр}}$ при наличии избытка явной теплоты должна быть на $5 - 7 \text{ } ^{\circ}\text{C}$ ниже температуры воздуха в рабочей зоне:

$$t_{\text{пр}} = 22 - 7 = 15 \text{ } ^{\circ}\text{C}$$

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

$$Q_{\text{изб}} = \sum Q - \sum Q_{\text{ух}} \text{ } ^{[15]} \quad (5.14)$$

где $\sum Q$ – суммарное количество поступающей в помещение явной теплоты;

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

Основными источниками избыточного тепла являются светильники, люди и др. Кроме того, необходимо учитывать тепlopоступления от

солнечной радиации. В данном помещении тепловыделением электронного оборудования можно пренебречь. Поэтому учитываем тепловыделения от искусственного освещения, от людей, количество тепла, поступающего в помещение через окна от солнечной радиации.

Тепловыделения от искусственного освещения Q_2 , рассчитывают, предполагая, что практически вся затрачиваемая энергия, в конечном счете, преобразуется в тепло, по формуле

$$Q_2 = 1000 \cdot N^{[15]} \quad (5.15)$$

где N – расходуемая мощность светильников, Вт.

$$Q_2 = 1000 \cdot 0,28 \cdot 2 = 480 \text{ Вт.}$$

Тепловыделения от людей Q_3 определяют по формуле:

$$Q_3 = n \cdot q_{\text{ч}}^{[15]} \quad (5.16)$$

где n – число работающих;

$q_{\text{ч}}$ – количество тепла, выделяемое одним человеком, представлено в Таблице 38.

Таблица 38 – Количество тепла, выделяемое одним человеком в зависимости от категории работ и температуры окружающей среды

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

$$Q_3 = 1 \cdot 145 = 145 \text{ Вт.}$$

Количество тепла, поступающего в помещение от солнечной радиации $Q_{\text{ост.рад.}}$, определяют по формуле:

$$Q_{\text{ост.рад.}} = F_{\text{ост.}} \cdot q_{\text{ост.}} \cdot A_{\text{ост.}}^{[15]} \quad (5.17)$$

для покрытий:

$$Q_{\text{п рад.}} = F_n \cdot q_n \cdot k_n^{[15]} \quad (5.18)$$

где $F_{\text{ост.}}$ и F_n – площадь поверхности и покрытия, м^2 ;
 $q_{\text{ост.}}$ и q_n – теплопоступления через 1м^2 поверхности остекления и поверхности покрытия, при коэффициенте теплопередачи, равном $1 \frac{\text{Вт}}{\text{м}^2 \text{ } ^\circ\text{C}}$;
 $A_{\text{ост.}}$ – коэффициент остекления;
 k_n – коэффициент теплопередачи покрытия, $1 \frac{\text{Вт}}{\text{м}^2 \text{ } ^\circ\text{C}}$.

Значение $q_{\text{ост.}}$ в зависимости от географической ориентации поверхности и характеристики окон или фонарей принимается в пределах 70 - 210, а коэффициента $A_{\text{ост.}}$ в зависимости от вида остекления и его солнцезащитных свойств - в пределах 0,25 - 1,25, средние значения теплопоступления от солнечной радиации через покрытие в зависимости от географической широты и вида покрытия принимают в пределах 6 - 24

$$F_{\text{ост.}} = 1,5 \cdot 1,2 \cdot 2 \cdot 3,6 \text{ м}^2$$

Окно рабочего помещения направлено на север, поэтому примем значение равным $150 \frac{\text{Вт}}{\text{м}^2 \text{ } ^\circ\text{C}}$. Примем $A_{\text{ост.}} = 0,55$, по формуле (5.17):

$$Q_{\text{ост.рад.}} = 3,6 \cdot 150 \cdot 0,55 = 297 \text{ Вт.}$$

Среднее значение теплопоступления для покрытия с учетом географической широты примем равным $Q_{\text{п рад.}} = 18 \text{ Вт.}$

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

$$Q_{\text{ух}} = \frac{\lambda - S \cdot (t_{\text{выт}} - t_{\text{пр}})^{[15]}}{\delta} \quad (5.19)$$

где λ – теплопроводность стен, $\frac{\text{Вт}}{\text{м}^2 \text{ } ^\circ\text{C}}$

S – площадь, м^2 ;

δ – толщина стен, м.

Стены рабочего помещения изготовлены из тяжелого бетона М600, теплопроводность которого равна $12 \frac{\text{Вт}}{\text{м}^2 \text{ } ^\circ\text{C}}$. Толщина стен $\delta = 0,5\text{м}$.

$$Q_{\text{ух}} = \frac{1,2 \cdot 24 \cdot (21,4 + 15)}{0,5} = 368,64 \text{ Вт.}$$

Вычислим суммарное количество поступающей в помещение явной теплоты:

$$\sum Q = Q_2 + Q_3 + Q_{\text{ост.рад}} + Q_{\text{п.р.р.}} \quad [15] \quad (5.20)$$

$$\sum Q = 480 + 145 + 297 + 18 = 940 \text{ Вт}$$

Так как расчет производится для летнего периода величина избыточного выделения явной теплоты равна:

$$Q_{\text{изб}} = 940 \text{ Вт.}$$

Вычислим количество приточного воздуха:

$$L_{\text{пр}} = \frac{940}{1 \cdot 1,2 \cdot (21,4 - 15)} = 122,4 \frac{\text{м}^3}{\text{ч}}$$

Чтобы обеспечивать расход воздуха $L=122.4 \text{ м}^3/\text{ч}$, можно использовать 1 кондиционер фирмы SAMSUNG MonteAQ-09AWA с максимальным расходом воздуха $195 \text{ м}^3/\text{ч}$.



Рисунок 5.3 – Кондиционер SAMSUNG MonteAQ-09AWA

Краткие характеристики модели SAMSUNG Monte AQ-09AWA:

- Тип: крышный моноблок промышленного кондиционера.
- Номинальная холодопроизводительность Вт 23200 .
- Номинальная потребляемая мощность Вт 9400. Расход воздуха (min-max) м³/ч 100-195.

Что соответствует действительности и является достаточным для обеспечения комфортного микроклимата.

Заключение

В процессе написания дипломного проекта мною была разработана база данных для компьютерного магазина, которое будет хранить необходимые записи. В качестве предпочитаемого СУБД было выбрано MySQL, так как оно является легким и удобным для понимания, богат по функционалу. Разработка шла при сопровождении CASE-средств Rational Rose и ERwin.

В ходе выполнения проекта были получены навыки проектирования, моделирования и разработки базы данных в MySQL.

Произведен расчет на затраты по разработке системы в котором было рассчитана общая стоимость программного продукта и расчет затрат. Данный проект будет востребован на рынке, так как хорошо структурированная модель базы данных гарантирует максимальную целостность и структурированность информации, а его цена близка к обще рыночной.

В разделе безопасности-жизнедеятельности освещены все вредные факторы за рабочим местом, которые могут нанести хоть и незначительный, но вред здоровью человека. Так же были произведены расчеты по мерам защиты от поражения электрическим током и по системе кондиционирования.

Конечным результатом стало полностью разработанная и заполненная база данных хранящаяся в СУБД MySQL. Все поставленные задачи были выполнены.

В дальнейшем база данных может быть расширена по мере необходимости для хранения записей о других товарах.

Список литературы

- 1 Вебсайт «Технодом» - <http://www.technodom.kz/>
- 2 Вебсайт «Disti» - <http://www.distikz/>
- 3 Орлов С.А. Технологии разработки программного обеспечения. Питер, 2002г. – 340 с.
- 4 Роберт Дж. Мюллер. Базы данных и UML. Проектирование. Лори, 2002г. – 420 с.
- 5 А. Якобсон, Г. Буч, Дж. Рамбо Унифицированный процесс разработки ПО, М., 2000 г. – 240 с.
- 6 Введение в UML. Лекции. [Электронный ресурс] – Адрес: <http://www.intuit.ru>. «Таблицы UML, обозначения» - 30.10.2010.
- 7 Эрик Дж. Нейбург, Роберт А. Максимчук. Проектирование баз данных с помощью UML. Вильемс, 2002. - 288 с.
- 8 Сатимова Е.Г. Проектирование баз данных: Методические указания к выполнению лабораторных работ. – Алматы: АИЭС, 2009. – 32 с.
- 9 Питер Роб, Карлос Коронелл. Система баз данных: проектирование, реализация и управление. – 5-е изд. – СПб.: БХВ - Петербург, 2004. - 854 с.
- 10 Кристиан Уэнц. PHP и MySQL. Карманный справочник, 2013. - 192 с.
- 11 Документация по Linux ОС CentOS [Электронный ресурс] – Адрес: <https://www.centos.org/> «Команды, обозначения»
- 12 Линн Бейли, Майкл Моррисон. Изучаем PHP и MySQL. – Москва, Эксмо, 2010. – 340 с.
- 13 Бретт Маклафлин. PHP и MySQL. Исчерпывающее руководство Питер, 2003. – 512 с.
- 14 Базылов К.Б., Алибаева С.А., Бабич А.А. Методические указания по выполнению экономического раздела выпускной работы бакалавров. – Алматы: АИЭС, 2008. – 28 с.
- 15 Бекишева А.И. Методические указания к выполнению экономической части дипломной работы для студентов специальности 5В070400 – ВТиПО – Алматы: АУЭС, 2014. – 42 с.
- 16 А.И. Рофе. Экономика труда: учебник. – М.: КиноРус, 2010. – 400 с.
- 17 СНиП РК 4.02–42–2006 Отопление, вентиляция и кондиционирование. – Астана: Издательство стандартов, 2007. – 218 с.
- 18 Белов С.В., Ильницкая А.В., Козьяков А.Ф. и др. Безопасность жизнедеятельности. Учебник для вузов. – М.: Высш.шк., 2005. – 448 с.
- 19 Хакимжанов Т.Е. Сборник задач по охране труда и безопасности жизнедеятельности: Учебное пособие для вузов. – Алматы: Эверо, 2007. – 274с.

Приложение А

```
CREATE TABLE client
```

```
(  
    cli_id          INTEGER NOT NULL PRIMARY KEY,  
    cli_login       VARCHAR(40) NULL,  
    cli_fname       VARCHAR(40) NULL,  
    cli_sname       VARCHAR(40) NULL,  
    cli_phone       INTEGER NULL,  
    cli_email       VARCHAR(40) NULL  
);
```

```
CREATE TABLE cl_order
```

```
(  
    or_id          INTEGER NOT NULL,  
    cli_id         INTEGER NOT NULL,  
    registration_date DATE NULL,  
    completion_date DATE NULL,  
    PRIMARY KEY (or_id,cli_id),  
    FOREIGN KEY R_1 (cli_id) REFERENCES client (cli_id)  
);
```

```
CREATE TABLE cpu
```

```
(  
    cpu_id         INTEGER NOT NULL PRIMARY KEY  
    AUTO_INCREMENT,  
    cpu_prod       VARCHAR(40) NULL,  
    cpu_type       VARCHAR(40) NULL,  
    cpu_freq_ghz   FLOAT NULL,  
    cpu_core       VARCHAR(40) NULL,  
    cpu_socket     VARCHAR(40) NULL,  
    cpu_price_tg   INTEGER NULL  
);
```

```
CREATE TABLE cpu_cooler
```

```
(  
    coo_id         INTEGER NOT NULL PRIMARY KEY  
    AUTO_INCREMENT,
```

```
    coo_prod      VARCHAR(40) NULL,  
    coo_model     VARCHAR(40) NULL,
```

Продолжение приложения А

```
    coo_socket    VARCHAR(40) NULL,  
    coo_rotation_rpm  INTEGER NULL,  
    coo_radiator  VARCHAR(40) NULL,  
    coo_price_tg  INTEGER NULL
```

```
);
```

```
CREATE TABLE hdd
```

```
(
```

```
    hdd_id        INTEGER NOT NULL PRIMARY KEY  
    AUTO_INCREMENT,  
    hdd_prod      VARCHAR(40) NULL,  
    hdd_model     VARCHAR(40) NULL,  
    hdd_capacity_gb  INTEGER NULL,  
    hdd_bufer_mb  INTEGER NULL,  
    hdd_rotation_rpm  INTEGER NULL,  
    hdd_format    FLOAT NULL,  
    hdd_interface VARCHAR(40) NULL,  
    hdd_price_tg  INTEGER NULL
```

```
);
```

```
CREATE TABLE motherboard
```

```
(
```

```
    mb_id         INTEGER NOT NULL PRIMARY KEY  
    AUTO_INCREMENT,  
    mb_type       VARCHAR(40) NULL,  
    mb_socket     VARCHAR(40) NULL,  
    mb_chipset    VARCHAR(40) NULL,  
    mb_maxRamMemory_gb  INTEGER NULL,  
    mb_bios       VARCHAR(40) NULL,  
    mb_ports      TEXT NULL,  
    mb_img        VARCHAR(255) NULL,  
    mb_price_tg   INTEGER NULL
```

```
);
```

```
CREATE TABLE pc_case
```

(

Продолжение приложения А

```
case_id      INTEGER NOT NULL PRIMARY KEY
AUTO_INCREMENT,
case_prod    VARCHAR(40) NULL,
case_model   VARCHAR(40) NULL,
case_format  VARCHAR(40) NULL,
case_material VARCHAR(40) NULL,
case_color   VARCHAR(40) NULL,
case_length_cm  FLOAT NULL,
case_height_cm  FLOAT NULL,
case_width_cm  FLOAT NULL,
case_img      VARCHAR(255) NULL,
case_price_tg  INTEGER NULL
);
```

CREATE TABLE power_supply

```
(
ps_id      INTEGER NOT NULL PRIMARY KEY
AUTO_INCREMENT,
ps_prod    VARCHAR(40) NULL,
ps_model   VARCHAR(40) NULL,
ps_power_w  INTEGER NULL,
ps_price_tg  INTEGER NULL
);
```

CREATE TABLE ram

```
(
ram_id      INTEGER NOT NULL PRIMARY KEY
AUTO_INCREMENT,
ram_prod    VARCHAR(40) NULL,
ram_model   VARCHAR(40) NULL,
ram_memSize_gb  INTEGER NULL,
ram_memType  VARCHAR(40) NULL,
ram_freq_mhz  FLOAT NULL,
ram_price_tg  INTEGER NULL
);
```

);

Продолжение приложения А

```
CREATE TABLE ssd
(
    ssd_id          INTEGER NOT NULL PRIMARY KEY
    AUTO_INCREMENT,
    ssd_prod        VARCHAR(40) NULL,
    ssd_model       VARCHAR(40) NULL,
    ssd_capacity_gb INTEGER NULL,
    ssd_format      FLOAT NULL,
    ssd_interface   VARCHAR(40) NULL,
    ssd_speedR_mbs  INTEGER NULL,
    ssd_speedW_mbs  INTEGER NULL,
    ssd_price_tg    INTEGER NULL
);
```

```
CREATE TABLE videocard
(
    vid_id          INTEGER NOT NULL PRIMARY KEY
    AUTO_INCREMENT,
    vid_prod        VARCHAR(40) NULL,
    vid_model       VARCHAR(40) NULL,
    vid_gpu         VARCHAR(40) NULL,
    vid_freq_mhz    FLOAT NULL,
    vid_api         TEXT NULL,
    vid_mem_byte    INTEGER NULL,
    vid_memory_gb   INTEGER NULL,
    vid_type        VARCHAR(40) NULL,
    vid_mem_freq_mhz FLOAT NULL,
    vid_price_tg    INTEGER NULL
);
```

```
CREATE TABLE pc
(
    pc_id          INTEGER NOT NULL PRIMARY KEY
    AUTO_INCREMENT,
```

```
cpu_id      INTEGER NULL,  
coo_id      INTEGER NULL,
```

Продолжение приложения А

```
case_id     INTEGER NULL,  
ps_id       INTEGER NULL,  
hdd_id      INTEGER NULL,  
ssd_id      INTEGER NULL,  
ram_id      INTEGER NULL,  
vid_id      INTEGER NULL,  
mb_id       INTEGER NULL,  
FOREIGN KEY R_9 (cpu_id) REFERENCES cpu (cpu_id),  
FOREIGN KEY R_10 (coo_id) REFERENCES cpu_cooler (coo_id),  
FOREIGN KEY R_11 (case_id) REFERENCES pc_case (case_id),  
FOREIGN KEY R_12 (ps_id) REFERENCES power_supply (ps_id),  
FOREIGN KEY R_13 (hdd_id) REFERENCES hdd (hdd_id),  
FOREIGN KEY R_14 (ssd_id) REFERENCES ssd (ssd_id),  
FOREIGN KEY R_15 (ram_id) REFERENCES ram (ram_id),  
FOREIGN KEY R_16 (vid_id) REFERENCES videocard (vid_id),  
FOREIGN KEY R_17 (mb_id) REFERENCES motherboard (mb_id)  
);
```

```
CREATE TABLE producer  
(  
    prod_id      INTEGER NOT NULL PRIMARY KEY,  
    prod_name    VARCHAR(40) NULL  
);
```

```
CREATE TABLE display  
(  
    dis_id       INTEGER NOT NULL PRIMARY KEY  
AUTO_INCREMENT,  
    dis_diag_inch  FLOAT NULL,  
    dis_res_x     INTEGER NULL,  
    dis_res_y     INTEGER NULL,  
    dis_matrix    VARCHAR(40) NULL,  
    dis_type      VARCHAR(40) NULL,  
    dis_sensor    INTEGER NULL
```

);

Продолжение приложения А

```
CREATE TABLE cpu_m
(
    m_cpu_id          INTEGER NOT NULL PRIMARY KEY
    AUTO_INCREMENT,
    prod_id           INTEGER NULL,
    m_cpu_model       VARCHAR(40) NULL,
    m_cpu_freqMin_ghz  FLOAT NULL,
    m_cpu_freqMax_ghz  FLOAT NULL,
    FOREIGN KEY R_26 (m_cpu_id) REFERENCES producer (prod_id)
);
```

```
CREATE TABLE ram_m
(
    m_ram_id          INTEGER NOT NULL PRIMARY KEY
    AUTO_INCREMENT,
    m_ram_model       VARCHAR(40) NULL,
    m_ram_freq_ghz    FLOAT NULL,
    m_ram_memory_gb   INTEGER NULL
);
```

```
CREATE TABLE storage
(
    m_stor_id         INTEGER NOT NULL PRIMARY KEY
    AUTO_INCREMENT,
    m_stor_type       VARCHAR(40) NULL,
    m_stor_memory_gb  INTEGER NULL,
);
```

```
CREATE TABLE videocard_m
(
    m_vid_id          INTEGER NOT NULL PRIMARY KEY
    AUTO_INCREMENT,
    prod_id           INTEGER NULL,
    m_vid_model       VARCHAR(40) NULL,
    m_vid_memory_gb   INTEGER NULL,
    m_vid_byte        INTEGER NULL,
);
```

```
FOREIGN KEY R_28 (m_vid_id) REFERENCES producer (prod_id)
);
```

Продолжение приложения А

```
CREATE TABLE os
(
    os_id          INTEGER NOT NULL PRIMARY KEY
AUTO_INCREMENT,
    prod_id        INTEGER NULL,
    os_name        VARCHAR(40) NULL,
    os_bit         INTEGER NULL,
    FOREIGN KEY R_29 (os_id) REFERENCES producer (prod_id)
);
```

```
CREATE TABLE ports
(
    m_po_id        INTEGER NOT NULL PRIMARY KEY
AUTO_INCREMENT,
    m_po_descrip   TEXT NULL
);
```

```
CREATE TABLE audio
(
    m_aud_id       INTEGER NOT NULL PRIMARY KEY
AUTO_INCREMENT,
    m_aud_technology VARCHAR(40) NULL
);
```

```
CREATE TABLE network_m
(
    m_net_id       INTEGER NOT NULL PRIMARY KEY
AUTO_INCREMENT,
    m_net_bluetooth FLOAT NULL,
    m_net_wifi     VARCHAR(20) NULL
);
```

```
CREATE TABLE device
(
    dev_id         INTEGER NOT NULL PRIMARY KEY
AUTO_INCREMENT,
```

dev_type VARCHAR(40) NULL,

Продолжение приложения А

dev_prod VARCHAR(40) NOT NULL,
dev_model VARCHAR(40) NULL,
m_cpu_id INTEGER NULL,
m_ram_id INTEGER NULL,
dis_id INTEGER NULL,
m_stor_id INTEGER NULL,
m_vid_id INTEGER NULL,
m_aud_id INTEGER NULL,
os_id INTEGER NULL,
m_po_id INTEGER NULL,
m_net_id INTEGER NULL,
dev_img VARCHAR(255) NULL,
dev_price_tg INTEGER NULL,
FOREIGN KEY R_30 (m_cpu_id) REFERENCES cpu_m (m_cpu_id),
FOREIGN KEY R_31 (m_ram_id) REFERENCES ram_m (m_ram_id),
FOREIGN KEY R_32 (dis_id) REFERENCES display (dis_id),
FOREIGN KEY R_33 (m_stor_id) REFERENCES storage (m_stor_id),
FOREIGN KEY R_34 (m_vid_id) REFERENCES videocard_m (m_vid_id),
FOREIGN KEY R_35 (m_aud_id) REFERENCES audio (m_aud_id),
FOREIGN KEY R_37 (os_id) REFERENCES os (os_id),
FOREIGN KEY R_38 (m_po_id) REFERENCES ports (m_po_id),
FOREIGN KEY R_39 (m_net_id) REFERENCES network_m (m_net_id)

);

CREATE TABLE periphery

(
 peri_id INTEGER NOT NULL PRIMARY KEY,
 peri_name VARCHAR(40) NULL,
 prod_id INTEGER NULL,
 peri_descrip TEXT NULL,
 peri_img VARCHAR(255) NULL,
 peri_price_tg INTEGER NULL,
 FOREIGN KEY R_25 (prod_id) REFERENCES producer (prod_id)

);


```
CREATE TABLE or_device
```

```
(
```

```
    or_id          INTEGER NULL,
```

Продолжение приложения А

```
    cli_id         INTEGER NULL,
```

```
    dev_id         INTEGER NULL,
```

```
    FOREIGN KEY R_2 (or_id, cli_id) REFERENCES cl_order (or_id, cli_id),
```

```
    FOREIGN KEY R_4 (dev_id) REFERENCES device (dev_id)
```

```
);
```

```
CREATE TABLE or_pc
```

```
(
```

```
    or_id          INTEGER NULL,
```

```
    cli_id         INTEGER NULL,
```

```
    pc_id          INTEGER NULL,
```

```
    FOREIGN KEY R_3 (or_id, cli_id) REFERENCES cl_order (or_id, cli_id),
```

```
    FOREIGN KEY R_5 (pc_id) REFERENCES pc (pc_id)
```

```
);
```

```
CREATE TABLE or_peri
```

```
(
```

```
    or_id          INTEGER NULL,
```

```
    cli_id         INTEGER NULL,
```

```
    peri_id        INTEGER NULL,
```

```
    FOREIGN KEY R_19 (or_id, cli_id) REFERENCES cl_order (or_id, cli_id),
```

```
    FOREIGN KEY R_21 (peri_id) REFERENCES periphery (peri_id)
```

```
);
```

Приложение Б

```
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 1,
"Unknown");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 2,
"Microsoft");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 3,
"Community");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 4,
"Intel");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 5,
"nVidia");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 6,
"AMD Radeon");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 7,
"Acer");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 8,
"Asus");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 9,
"Apple");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 10,
"HP");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 11,
"Lenovo");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 12,
"Packard Bell");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 13,
"MSI");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 14,
"Dell");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 15,
"Samsung");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 16,
"Sony");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 17,
"Kingston");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 18,
"Seagate");
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 19, "A4
Tech");
```

Продолжение приложения Б

```
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 20, "Fujitsu");
```

```
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 21, "Western Digital");
```

```
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 22, "Silicon Power");
```

```
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 23, "Transcend");
```

```
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 24, "BenQ");
```

```
INSERT INTO `PRODUCER` (`prod_id`, `prod_name`) VALUES ( 25, "Hitachi");
```

```
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`, `m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)
```

```
VALUES ( 4, "Celeron N2840", 2.167, 2.58);
```

```
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`, `m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)
```

```
VALUES ( 4, "Celeron N3050", 1.6, 2.2);
```

```
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`, `m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)
```

```
VALUES ( 4, "Atom x5 Z8500", 1.44, 2.24);
```

```
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`, `m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)
```

```
VALUES ( 4, "Pentium N3700", 1.6, 2.4);
```

```
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`, `m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)
```

```
VALUES ( 4, "Core i3-4005U", 1.7, 2.3);
```

```
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`, `m_cpu_freqMin_ghz`)
```

```
VALUES ( 4, "Core i3-4000M", 2.4);
```

```
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`, `m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)
```

```
VALUES ( 4, "Core i5-5200U", 2.2, 2.5);
```

```
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`, `m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)
```

```
VALUES ( 4, "Core i5-6200U", 2.3, 2.8);
```

```
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)
```

Продолжение приложения Б

```
VALUES ( 4, "Core i5-540M", 2.53, 3.1);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core i7-4720HQ", 2.6, 3.6);  
11 INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core i7", 2.5, 3.7);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core i7-3820QM", 2.7, 3.7);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core M-70", 1.1, 2.6);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core i5", 3.2, 3.6);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core i5-5675R", 3.1, 3.6);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`)  
VALUES ( 4, "Core i3-5010U", 2.1);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core i7-6700T", 2.8, 3.6);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core i5-3470S", 2.9, 3.9);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core i7-4500U", 1.8, 3.0);  
INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`, `m_cpu_freqMax_ghz`)  
VALUES ( 4, "Core i5-6400T", 2.2, 2.8);  
21 INSERT INTO `CPU_M` (`prod_id`, `m_cpu_model`,  
`m_cpu_freqMin_ghz`)
```

```
VALUES ( 4, "Pentium G2030", 3.0);
```

Продолжение приложения Б

```
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3L", 1.3, 4);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3L", 1.6, 2);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3", 1.6, 4);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3", 1.6, 2);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3L", 1.6, 4);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3", 1.6, 6);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3", 1.1, 4);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3L", 1.6, 8);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3", 1.6, 16);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("DDR3", 1.8, 8);  
11 INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)  
VALUES ("LPDDR3", 1.8, 8);  
INSERT INTO `RAM_M` (`m_ram_model`, `m_ram_freq_ghz`,  
`m_ram_memory_gb`)
```

```
VALUES ("DDR3", 1.6, 8);
```

Продолжение приложения Б

```
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 15.6, 1366, 768, "IPS", "HD", 0);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 11.6, 1366, 768, "IPS", "HD", 0);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 10.1, 1280, 800, "IPS", "WXGA", 1);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 14.0, 1366, 768, "IPS", "HD", 0);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 17.3, 1920, 1080, "IPS", "FullHD", 0);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 13.3, 1366, 768, "IPS", "HD", 0);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 17, 1920, 1200, "WUXGA", "FullHD", 0);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 15.4, 2880, 1800, "IPS", "FullHD", 0);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 18.4, 1920, 1080, "IPS", "FullHD", 0);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 13.3, 3200, 1800, "IPS+IGZO", "QHD+", 1);  
11 INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)  
VALUES ( 27, 5120, 2880, "IPS", "5k", 0);  
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,  
`dis_matrix`, `dis_type`, `dis_sensor`)
```

```
VALUES ( 21.5, 4096, 2304, "IPS", "4k", 0);
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,
`dis_matrix`, `dis_type`, `dis_sensor`)
```

Продолжение приложения Б

```
VALUES ( 23, 1920, 1080, "IPS", "FullHD", 1);
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,
`dis_matrix`, `dis_type`, `dis_sensor`)
VALUES ( 27, 2560, 1440, "IPS", "FullHD", 1);
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,
`dis_matrix`, `dis_type`, `dis_sensor`)
VALUES ( 27, 1920, 1080, "IPS", "FullHD", 1);
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,
`dis_matrix`, `dis_type`, `dis_sensor`)
VALUES ( 21.5, 1920, 1080, "IPS", "FullHD", 0);
INSERT INTO `DISPLAY` (`dis_diag_inch`, `dis_res_x`, `dis_res_y`,
`dis_matrix`, `dis_type`, `dis_sensor`)
VALUES ( 19.5, 1600, 900, "IPS", "FullHD", 1);
```

```
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("HDD", 500);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("eMMC", 32);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("HDD", 1000);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("eMMC", 64);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("HDD", 750);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("HDD+SSD", 628);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("SSD", 512);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("HDD+SSD", 1256);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("SSD", 256);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("HDD+SSD", 2128);
11 INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
```

```
VALUES ("HDD", 2000);
INSERT INTO `STORAGE` (`m_stor_type`, `m_stor_memory_gb`)
VALUES ("HDD+SSD", 1128);
```

Продолжение приложения Б

```
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 4, "HD Graphics", 1, 16);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 5, "GeForce GT 810M", 1, 8);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 4, "HD Graphics 4400", 1, 16);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 5, "GeForce GT 940M", 2, 16);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 5, "GeForce GT 740M", 2, 16);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 4, "HD Graphics 5500", 1, 16);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 4, "HD Graphics 520", 1, 16);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 5, "GeForce GTX 970M", 3, 32);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 6, "Radeon R9 M370X", 2, 16);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 5, "GeForce GTX 675M", 2, 32);
11 INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
VALUES ( 4, "HD Graphics 5300", 1, 16);
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,
`m_vid_memory_gb`, `m_vid_byte`)
```



```
VALUES ( 6, "Radeon R9 M380", 2, 16);
```

Продолжение приложения Б

```
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,  
`m_vid_memory_gb`, `m_vid_byte`)
```

```
VALUES ( 4, "Iris Pro Graphics 6200", 1, 16);
```

```
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,  
`m_vid_memory_gb`, `m_vid_byte`)
```

```
VALUES ( 6, "Radeon R7 A365", 4, 16);
```

```
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,  
`m_vid_memory_gb`, `m_vid_byte`)
```

```
VALUES ( 5, "GeForce GT 630M", 2, 16);
```

```
INSERT INTO `VIDEOCARD_M` (`prod_id`, `m_vid_model`,  
`m_vid_memory_gb`, `m_vid_byte`)
```

```
VALUES ( 5, "GeForce 840M", 1, 16);
```

```
INSERT INTO `OS` (`prod_id`, `os_name`, `os_bit`)
```

```
VALUES ( 2, "Windows 7", 64);
```

```
INSERT INTO `OS` (`prod_id`, `os_name`, `os_bit`)
```

```
VALUES ( 2, "Windows 10", 64);
```

```
INSERT INTO `OS` (`prod_id`, `os_name`, `os_bit`)
```

```
VALUES ( 2, "Windows 8", 64);
```

```
INSERT INTO `OS` (`prod_id`, `os_name`, `os_bit`)
```

```
VALUES ( 9, "Mac OS X 10.6 Snow Leopard", 64);
```

```
INSERT INTO `OS` (`prod_id`, `os_name`, `os_bit`)
```

```
VALUES ( 9, "Mac OS X Yosemite", 64);
```

```
INSERT INTO `OS` (`prod_id`, `os_name`, `os_bit`)
```

```
VALUES ( 9, "Mac OS X El Capitan", 64);
```

```
INSERT INTO `OS` (`prod_id`, `os_name`, `os_bit`)
```

```
VALUES ( 2, "Windows 8.1", 64);
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("1x USB 2.0 / 1x USB 3.0 / 1x RJ45 / 1x HDMI / 1x SD слот / 1x  
COMBO audio jack / 1x Kensington lock");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("1x HDMI / 1x USB 3.0 / 1x USB / 1x audio jack / 1x MicroSD  
слот / 1x Kensington lock");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("1x audio jack / 1x D-Sub / 1x USB 3.0 / 1x USB 2.0 / 1x USB  
типа C (up to 5 Gbps) / 1x RJ45 / 1x HDMI / 1x SD слот / 1x Kensington lock");  
INSERT INTO `PORTS` (`m_po_descrip`)
```

Продолжение приложения Б

```
VALUES ("1x audio jack / 1x USB 2.0 / 1x USB Type C / 1x MicroUSB / 1x  
MicroHDMI / 1x MicroSD");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("1x audio jack / 1x D-Sub / 1x USB 3.0 / 1x USB 2.0 / 1x USB  
Type C / 1x RJ45 / 1x HDMI / 1x SD слот / 1x Kensington lock");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("1x audio jack / 1x USB 2.0 / 1x USB 3.0 / 1x D-Sub / 1x HDMI /  
1x RJ45 / 1x SD слот / 1x Kensington lock");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("1x HDMI / 2x USB 2.0 / 1x USB 3.0 / 1x RJ45 / 1x audio jack /  
1x SD слот / 1x Kensington lock");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("1x RJ45 / 1x HDMI / 1x D-Sub / 1x USB 2.0 / 2x USB 3.0 / 1x  
SD слот / 1x COMBO jack / 1x Kensington lock");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("1x RJ45 / 1x HDMI / 4x USB 3.0 / 1x SD слот / 1x COMBO  
jack / 1x Kensington lock");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("2x USB 2.0 / 1x USB 3.0 / 1x RJ45 / 1x D-Sub / 1x HDMI / 1x  
SD слот / 1x COMBO audio jack / 1x Kensington lock");
```

```
11 INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("3x USB 2.0 / Mini DisplayPort / IEEE1394 / RJ-45 / 2x audio  
jack / 1x Kensington lock");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("1x HDMI / 4x USB 3.0 / 1x RJ45 / 1x D-Sub / 1x SD слот / 1x  
Thunderbolt порт / 1x разъем под наушники (совмещен с SPDIF) / 1x разъем  
под микрофон / 1x AC адаптер питания / 1x Kensington lock");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("2x USB 3.0 / HDMI / 2x Mini DisplayPort(Thunderbolt) / 1x  
audio jack");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)  
VALUES ("4x USB 3.0 / USB2.0 (E-SATA) / HDMI / Mini DisplayPort /  
RJ-45 / VGA / 1x audio jack / S-PDIF / HDMI in");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("2x USB 3.0 / USB 2.0 / AC / Card Reader / Micro HDMI / MiniJack");  
INSERT INTO `PORTS` (`m_po_descrip`)
```

Продолжение приложения Б

```
VALUES ("4x USB 3.0 / 2x Mini DisplayPort (Thunderbolt) / RJ-45 / 1x audio jack / SDXC / SDHC / SD");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("4x USB 3.0 / 2x USB 2.0 / HDMI in / HDMI / RJ-45 / 1x audio jack / Subwoofer");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("4x USB 3.0 / 2x USB 3.0 / HDMI in / HDMI / RJ-45 / 1x audio jack / SDHC / SD / MMC / MS");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("4x USB 2.0 / HDMI in / HDMI / RJ-45 / Линейный выход / 1x audio jack / SDHC / SD / MMC / MS");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("1x USB 3.0 / 2x USB 2.0 / HDMI in / RJ-45 / 1x audio jack / SDHC / SD");
```

```
21 INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("2x USB 3.0 / 1x audio jack");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("4x USB 3.0 / 2x USB 2.0 / HDMI in / HDMI / RJ-45 / 1x audio jack / Subwoofer / SDXC / SDHC / SD / MMC / MS Pro / MS");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("4x USB 3.0 / 2x USB 3.0 / HDMI in / HDMI / RJ-45 / 1x audio jack / SDXC / SDHC / SD");
```

```
INSERT INTO `PORTS` (`m_po_descrip`)
```

```
VALUES ("4x USB 2.0 / RJ-45 / HDMI / HDMI in / 1x audio jack / SDHC / SD / MMC / MS");
```

```
INSERT INTO `AUDIO` (`m_aud_technology`)
```

```
VALUES ("Usual");
```

```
INSERT INTO `AUDIO` (`m_aud_technology`)
```

```
VALUES ("ICESound");
```

```
INSERT INTO `AUDIO` (`m_aud_technology`)
```

```
VALUES ("SonicMaster");
```

```
INSERT INTO `AUDIO` (`m_aud_technology`)
```

```
VALUES ("Waves MaxxAudio 4");
```

```
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("Beats");
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("Dolby Advanced Audio");
```

Продолжение приложения Б

```
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("DTS Studio Sound");
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("Double Layer");
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("MaxxAudio");
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("Klipsch");
11 INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("Waves Audio");
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("Cirrus Logic");
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("Bang & Olufsen");
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("THX TruStudio PRO");
INSERT INTO `AUDIO` (`m_aud_technology`)
VALUES ("Dolby Home Theater v4");
```

```
INSERT INTO `NETWORK_M` (`m_net_bluetooth`, `m_net_wifi`)
VALUES ( 4.02, "802.11 b/g/n");
INSERT INTO `NETWORK_M` (`m_net_bluetooth`, `m_net_wifi`)
VALUES ( 4.0, "802.11 b/g/n");
INSERT INTO `NETWORK_M` (`m_net_bluetooth`, `m_net_wifi`)
VALUES ( 4.0, "802.11 a/b/g/n");
INSERT INTO `NETWORK_M` (`m_net_bluetooth`, `m_net_wifi`)
VALUES ( 4.0, "802.11 a/b/g/n/ac");
INSERT INTO `NETWORK_M` (`m_net_bluetooth`, `m_net_wifi`)
VALUES ( 2.1, "802.11 a/b/g");
INSERT INTO `NETWORK_M` (`m_net_bluetooth`, `m_net_wifi`)
VALUES ( 3.0, "802.11 b/g/n");
INSERT INTO `NETWORK_M` (`m_net_bluetooth`, `m_net_wifi`)
VALUES ( 4.0, "802.11 a/c");
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

Продолжение приложения Б

```
VALUES ("Laptop", "Lenovo", "B5010 (80QR003BRK)", 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, "img_path", 76990);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Laptop", "Dell", "Inspiron 3162", 2, 2, 2, 2, 1, 1, 2, 2, 2, "img_path", 79990);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Laptop", "Asus", "X540SA-XX009D", 2, 3, 1, 3, 1, 2, 1, 3, 2, "img_path", 93384);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Laptop", "Asus", "Transformer Book T100HA", 3, 4, 3, 4, 1, 1, 2, 4, 3, "img_path", 131477);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Laptop", "Asus", "X540SC-XX015T", 4, 2, 1, 1, 2, 3, 2, 5, 2, "img_path", 108990);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Laptop", "Asus", "E402SA-WX034D", 4, 3, 4, 1, 1, 3, 1, 6, 2, "img_path", 109990);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Laptop", "Dell", "Inspiron 3542", 5, 5, 1, 1, 3, 4, 2, 7, 2, "img_path", 144990);
```

```

INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
VALUES ("Laptop", "Asus", "X555LB-XO487T", 5, 5, 1, 3, 4, 1, 2, 8, 2,
"img_path", 170990);

```

Продолжение приложения Б

```

INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
VALUES ("Laptop", "HP", "Envy 17-j010er", 6, 6, 5, 5, 5, 5, 3, 9, 2,
"img_path", 240990);

```

```

INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
VALUES ("Laptop", "Lenovo", "G5080 (80E502RLRK)", 7, 5, 1, 3, 6, 6, 3,
10, 2, "img_path", 168990);

```

```

INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
VALUES ("Laptop", "HP", "ProBook 430 G3 P5S49EA", 8, 3, 6, 6, 7, 7, 2,
11, 4, "img_path", 188990);

```

```

INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
VALUES ("Laptop", "Apple", "MacBook Pro 17 (A1297)", 9, 7, 7, 1, 7, 8, 4,
12, 5, "img_path", 560299);

```

```

INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
VALUES ("Laptop", "Asus", "G751JT-T7155T", 10, 8, 5, 3, 8, 9, 2, 13, 4,
"img_path", 564431);

```

```

INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
VALUES ("Laptop", "Apple", "MacBook Pro 15 Retina ", 11, 9, 8, 7, 9, 1, 5,
14, 4, "img_path", 699999);

```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Laptop", "Dell", "Alienware M18x", 12, 9, 9, 8, 10, 10, 2, 14, 6,
"img_path", 680000);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

Продолжение приложения Б

```
VALUES ("Laptop", "Lenovo", "Yoga 3 Pro", 13, 8, 10, 9, 11, 11, 2, 15, 7,
"img_path", 500000);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "Apple", "iMac Retina 5K 27 A1419
(MK462RU/A)", 14, 10, 11, 3, 12, 12, 6, 16, 4, "img_path", 949950);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "Apple", "iMac 21.5 Retina 4K A1418
(MK452RU/A)", 15, 11, 12, 3, 13, 12, 6, 16, 4, "img_path", 568690);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "Asus", "All-in-one ET2323IUK", 16, 3, 13, 3, 6, 1,
7, 17, 2, "img_path", 340000);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "HP", "Envy All-in-One 27-p001ur", 17, 9, 14, 10,
14, 13, 2, 18, 3, "img_path", 680135);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "MSI", "Wind Top AE2712G", 18, 12, 15, 3, 15,
14, 2, 19, 2, "img_path", 450000);
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "Acer", "Aspire Z1-623", 5, 7, 16, 1, 3, 1, 2, 20, 4,

```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "Lenovo", "IdeaCentre Flex 20", 19, 12, 17, 1, 3,
15, 7, 21, 2, 
```

Продолжение приложения Б

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "Asus", "All-in-one ET2323INT", 7, 6, 13, 11, 16,
3, 7, 22, 2, 
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "HP", "Envy 27-p000ur All-in-One", 20, 10, 14,
12, 14, 13, 2, 23, 3, 
```

```
INSERT INTO `DEVICE` (`dev_type`, `dev_prod`, `dev_model`,
`m_cpu_id`, `m_ram_id`, `dis_id`, `m_stor_id`, `m_vid_id`, `m_aud_id`, `os_id`,
`m_po_id`, `m_net_id`, `dev_img`, `dev_price_tg`)
```

```
VALUES ("Monoblock", "MSI", "Wind Top AE2282", 21, 3, 16, 1, 1, 1, 1,
24, 2, 
```

```
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
```

```
( "Intel", "Core i3-4330", 3.5, "Haswell", "LGA1150", 50000);
```

```
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
```

```
( "Intel", "Core i5-4460", 3.4, "Haswell Refresh", "LGA1150", 65000);
```

```
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
```

```
( "AMD", "FX-8320", 3.5, "Vishera", "AM3+", 58000);
```

```
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
```

```
( "Intel", "Quad Core", 2.3, "Kentsfield", "LGA775", 25000);
```

```
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
```



```

("Intel", "Celeron G1820", 2.7, "Haswell", "LGA1150", 11841);
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
("Intel", "Celeron G1840", 2.8, "Haswell", "LGA1150", 12260);
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
("AMD", "FX-4330", 4.0, "Vishera", "AM3+", 28581);

```

Продолжение приложения Б

```

INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
("Intel", "Core i5-4690", 3.5, "Haswell", "LGA1150", 75859);
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
("Intel", "Core i7-5960X", 3.0, "Haswell", "LGA2011", 432398);
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
("Intel", "Core i7-4790", 3.6, "Haswell", "LGA1150", 111430);
INSERT INTO `cpu` (`cpu_prod`, `cpu_type`, `cpu_freq_ghz`, `cpu_core`,
`cpu_socket`, `cpu_price_tg`) VALUES
("Intel", "Core i5-6600K", 3.5, "Skylake", "LGA1151", 91677);

```

```

INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("DeepCool", "Theta 115", "LGA1156/LGA1150/LGA1155", 1600,
"Aluminum", 1400);
INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("GlacialStars", "IceHut", "LGA1156", 2100, "Aluminum", 1390);
INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("CoolerMaster", "DKM-00001-A1-GP", "AM1", 4800, "Aluminum", 1600);
INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("DeepCool", "Beta 10", "AMD Hydro Bearing", 2200, "Aluminum", 853);
INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("DeepCool", "Alta 9", "LGA1150/LGA1155/LGA1156/LGA775", 2200,
"Aluminum", 1288);

```

```

INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("GlacialStars", "IceHut", "LGA1156", 1500, "Aluminum", 1324);
INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("Glacialtech", "Igloo 1100 PWM (E)", "LGA1156/LGA1155/LGA1150",
3600, "Aluminum", 2241);

```

Продолжение приложения Б

```

INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("DeepCool", "Beta 40", "AMD Hydro Bearing", 2200, "Aluminum", 3000);
INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("DeepCool", "Beta 200 ST", "AMD Hydro Bearing", 2200, "Aluminum",
3600);
INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("DeepCool", "Lucifer K2", "Intel LGA2011-
v3/LGA2011/LGA1366/LGA115X/LGA775
AMD FM2+/FM2/FM1/AM3+/AM3/AM2+/AM2", 1800,
"Copper/Aluminum", 15396);
INSERT INTO `cpu_cooler` (`coo_prod`, `coo_model`, `coo_socket`,
`coo_rotation_rpm`, `coo_radiator`, `coo_price_tg`) VALUES
("DeepCool", "Assassin II", "Intel Socket LGA2011-v3/ LGA2011/
LGA1366/ LGA1156/ LGA1155/ LGA1151/ LGA1150/ LGA775
AMD Socket FM2+/ FM2/ FM1/ AM3+/ AM3/ AM2+/ AM2", 1400,
"Copper/Aluminum", 26174);

```

```

INSERT INTO `pc_case` (`case_prod`, `case_model`, `case_format`,
`case_material`, `case_color`, `case_length_cm`, `case_height_cm`,
`case_width_cm`, `case_img`, `case_price_tg`) VALUES
("AeroCool", "CS-1102", "MiniTower", "Steel", "Black", 19.5, 41, 42.3,
"img_path", 9000);
INSERT INTO `pc_case` (`case_prod`, `case_model`, `case_format`,
`case_material`, `case_color`, `case_length_cm`, `case_height_cm`,
`case_width_cm`, `case_img`, `case_price_tg`) VALUES
("DeepCool", "Smarter", "MidiTower", "Steel", "Black/Red", 42, 20.1, 36.5,
"img_path", 9570);

```

```
INSERT INTO `pc_case` (`case_prod`, `case_model`, `case_format`,
`case_material`, `case_color`, `case_length_cm`, `case_height_cm`,
`case_width_cm`, `case_img`, `case_price_tg`) VALUES
("CoolerMaster", "HAF 922", "FullTower", "Steel", "Red/Black", 25.3, 50.2,
56.3, "img_path", 31000);
```

```
INSERT INTO `pc_case` (`case_prod`, `case_model`, `case_format`,
`case_material`, `case_color`, `case_length_cm`, `case_height_cm`,
`case_width_cm`, `case_img`, `case_price_tg`) VALUES
```

Продолжение приложения Б

```
("Delux", "DLC-MV871PSD", "MidiTower", "Steel", "Black", 18, 38, 40.8,
"img_path", 10000);
```

```
INSERT INTO `pc_case` (`case_prod`, `case_model`, `case_format`,
`case_material`, `case_color`, `case_length_cm`, `case_height_cm`,
`case_width_cm`, `case_img`, `case_price_tg`) VALUES
("GoldenField", "Aresze 7625B", "FullTower", "Steel", "Black", 48.9, 21.1,
50.7, "img_path", 10755);
```

```
INSERT INTO `pc_case` (`case_prod`, `case_model`, `case_format`,
`case_material`, `case_color`, `case_length_cm`, `case_height_cm`,
`case_width_cm`, `case_img`, `case_price_tg`) VALUES
("GoldenField", "Aresze 7209B", "MidiTower", "Steel", "Black", 21.1, 45,
48, "img_path", 10755);
```

```
INSERT INTO `power_supply` (`ps_prod`, `ps_model`, `ps_power_w`,
`ps_price_tg`) VALUES
("HuntKey", "UP-500", 500, 6300);
```

```
INSERT INTO `power_supply` (`ps_prod`, `ps_model`, `ps_power_w`,
`ps_price_tg`) VALUES
("AeroCool", "VX-350", 350, 6500);
```

```
INSERT INTO `power_supply` (`ps_prod`, `ps_model`, `ps_power_w`,
`ps_price_tg`) VALUES
("Delux", "DLW-550PA", 550, 15000);
```

```
INSERT INTO `power_supply` (`ps_prod`, `ps_model`, `ps_power_w`,
`ps_price_tg`) VALUES
("AeroCool", "SX-400", 240, 8500);
```

```
INSERT INTO `power_supply` (`ps_prod`, `ps_model`, `ps_power_w`,
`ps_price_tg`) VALUES
("HuntKey", "CP-450H", 450, 10500);
```

```
INSERT INTO `power_supply` (`ps_prod`, `ps_model`, `ps_power_w`,
`ps_price_tg`) VALUES
```

```
( "AeroCool", "VX-750", 750, 20000);
```

```
INSERT INTO `hdd` (`hdd_prod`, `hdd_model`, `hdd_capacity_gb`,  
`hdd_bufer_mb`, `hdd_rotation_rpm`, `hdd_format`, `hdd_interface`,  
`hdd_price_tg`) VALUES  
( "Toshiba", "DT01ACA100", 1000, 32, 7200, 3.5, "SATA 6 Gb/s", 18500);
```

Продолжение приложения Б

```
INSERT INTO `hdd` (`hdd_prod`, `hdd_model`, `hdd_capacity_gb`,  
`hdd_bufer_mb`, `hdd_rotation_rpm`, `hdd_format`, `hdd_interface`,  
`hdd_price_tg`) VALUES  
( "Western Digital", "Caviar Green WD30EZR", 3000, 64, 0, 3.5, "SATA 6  
Gb/s", 40000);
```

```
INSERT INTO `hdd` (`hdd_prod`, `hdd_model`, `hdd_capacity_gb`,  
`hdd_bufer_mb`, `hdd_rotation_rpm`, `hdd_format`, `hdd_interface`,  
`hdd_price_tg`) VALUES  
( "Seagate", "ST600MM0006", 600, 64, 10000, 2.5, "SAS 2.0", 43840);
```

```
INSERT INTO `hdd` (`hdd_prod`, `hdd_model`, `hdd_capacity_gb`,  
`hdd_bufer_mb`, `hdd_rotation_rpm`, `hdd_format`, `hdd_interface`,  
`hdd_price_tg`) VALUES  
( "Western Digital", "Caviar Green WD5000AZRX", 500, 64, 6800, 3.5,  
"SATA 6Gb/s", 18500);
```

```
INSERT INTO `hdd` (`hdd_prod`, `hdd_model`, `hdd_capacity_gb`,  
`hdd_bufer_mb`, `hdd_rotation_rpm`, `hdd_format`, `hdd_interface`,  
`hdd_price_tg`) VALUES  
( "Seagate", "Barracuda 7200.14 ST3000DM001", 2000, 64, 7200, 3.5,  
"SATA 6Gb/s", 36000);
```

```
INSERT INTO `ssd` (`ssd_prod`, `ssd_model`, `ssd_capacity_gb`,  
`ssd_format`, `ssd_interface`, `ssd_speedR_mbs`, `ssd_speedW_mbs`,  
`ssd_price_tg`) VALUES  
( "Silicon Power", "S55 SP120GBSS3S55S25", 120, 2.5, "SATA 6 Gb/s",  
550, 440, 22200);
```

```
INSERT INTO `ssd` (`ssd_prod`, `ssd_model`, `ssd_capacity_gb`,  
`ssd_format`, `ssd_interface`, `ssd_speedR_mbs`, `ssd_speedW_mbs`,  
`ssd_price_tg`) VALUES  
( "Plextor", "PX-128M6V", 128, 2.5, "DDR 2.0", 535, 170, 25000);
```

```

INSERT INTO `ssd` (`ssd_prod`, `ssd_model`, `ssd_capacity_gb`,
`ssd_format`, `ssd_interface`, `ssd_speedR_mbs`, `ssd_speedW_mbs`,
`ssd_price_tg`) VALUES
( "Kingston", "SV300S3D7/480G", 480, 2.5, "SATA 6 Gb/s", 450, 450,
65000);

```

```

INSERT INTO `ram` (`ram_prod`, `ram_model`, `ram_memSize_gb`,
`ram_memType`, `ram_freq_mhz`, `ram_price_tg`) VALUES
( "Kingston", "KVR16N11S8/4", 4, "DDR3", 1600, 7100);

```

Продолжение приложения Б

```

INSERT INTO `ram` (`ram_prod`, `ram_model`, `ram_memSize_gb`,
`ram_memType`, `ram_freq_mhz`, `ram_price_tg`) VALUES
( "Kingston", "HX318C10F/H", 4, "DDR3", 1866, 8500);

```

```

INSERT INTO `ram` (`ram_prod`, `ram_model`, `ram_memSize_gb`,
`ram_memType`, `ram_freq_mhz`, `ram_price_tg`) VALUES
( "Kingston", "HX321C11SR/8", 8, "DDR3", 2133, 20000);

```

```

INSERT INTO `ram` (`ram_prod`, `ram_model`, `ram_memSize_gb`,
`ram_memType`, `ram_freq_mhz`, `ram_price_tg`) VALUES
( "Kingston", "KVR16N11/8", 8, "DDR3", 1600, 12600);

```

```

INSERT INTO `ram` (`ram_prod`, `ram_model`, `ram_memSize_gb`,
`ram_memType`, `ram_freq_mhz`, `ram_price_tg`) VALUES
( "Kingston", "HyperX Savage (HX318C9SR/8)", 8, "DDR3", 1866, 15800);

```

```

INSERT INTO `ram` (`ram_prod`, `ram_model`, `ram_memSize_gb`,
`ram_memType`, `ram_freq_mhz`, `ram_price_tg`) VALUES
( "Kingston", "HyperX Savage (HX324C11SRK2/8)", 8, "DDR3", 2400,
25000);

```

```

INSERT INTO `videocard` (`vid_prod`, `vid_model`, `vid_gpu`,
`vid_freq_mhz`, `vid_api`, `vid_mem_byte`, `vid_memory_gb`, `vid_type`,
`vid_mem_freq_mhz`, `vid_price_tg`) VALUES

```

```

( "MSI", "N730K-2GD3/OC", "GeForce GT 730", 1006, "DirectX 12,
OpenGL 4.x", 8, 2, "DDR3", 900, 25000);

```

```

INSERT INTO `videocard` (`vid_prod`, `vid_model`, `vid_gpu`,
`vid_freq_mhz`, `vid_api`, `vid_mem_byte`, `vid_memory_gb`, `vid_type`,
`vid_mem_freq_mhz`, `vid_price_tg`) VALUES

```

```

( "MSI", "GeForce GTX950 GAMING 2G", "GeForce GTX 950", 1127,
"DirectX 12, OpenGL 4.x", 16, 2, "GDDR5", 1650, 75000);

```

```

INSERT INTO `videocard` (`vid_prod`, `vid_model`, `vid_gpu`,
`vid_freq_mhz`, `vid_api`, `vid_mem_byte`, `vid_memory_gb`, `vid_type`,
`vid_mem_freq_mhz`, `vid_price_tg`) VALUES
( "Gigabyte", "GV-N980G1 GAMING-4GD", "GeForce GTX 980", 1228,
"DirectX 11, OpenGL 4.x", 32, 4, "GDDR5", 1750, 215000);
INSERT INTO `videocard` (`vid_prod`, `vid_model`, `vid_gpu`,
`vid_freq_mhz`, `vid_api`, `vid_mem_byte`, `vid_memory_gb`, `vid_type`,
`vid_mem_freq_mhz`, `vid_price_tg`) VALUES
( "NVIDIA", "GeForce GTX 460", "GTX 460", 675, "DirectX 11, OpenGL
4.0", 32, 1, "GDDR5", 900, 178000);

```

Продолжение приложения Б

```

INSERT INTO `videocard` (`vid_prod`, `vid_model`, `vid_gpu`,
`vid_freq_mhz`, `vid_api`, `vid_mem_byte`, `vid_memory_gb`, `vid_type`,
`vid_mem_freq_mhz`, `vid_price_tg`) VALUES
( "Asus", "GTX Titan X", "Titan X", 1000, "DirectX 12, OpenGL 4.4", 8, 12,
"GDDR5", 1753, 430813);
INSERT INTO `videocard` (`vid_prod`, `vid_model`, `vid_gpu`,
`vid_freq_mhz`, `vid_api`, `vid_mem_byte`, `vid_memory_gb`, `vid_type`,
`vid_mem_freq_mhz`, `vid_price_tg`) VALUES
( "Asus", "MATRIX-GTX980Ti-6GD5-GAMING", "GTX980Ti", 1000,
"DirectX 12, OpenGL 4.5", 48, 6, "GDDR5", 1753, 284055);
INSERT INTO `videocard` (`vid_prod`, `vid_model`, `vid_gpu`,
`vid_freq_mhz`, `vid_api`, `vid_mem_byte`, `vid_memory_gb`, `vid_type`,
`vid_mem_freq_mhz`, `vid_price_tg`) VALUES
( "Palit", "nVidia GeForce GT740 (NEAT7400HD01)", "GeForce GT 740",
993, "DirectX 12, OpenGL 4.5", 16, 1, "DDR3", 1782, 27545);
INSERT INTO `videocard` (`vid_prod`, `vid_model`, `vid_gpu`,
`vid_freq_mhz`, `vid_api`, `vid_mem_byte`, `vid_memory_gb`, `vid_type`,
`vid_mem_freq_mhz`, `vid_price_tg`) VALUES
( "Gigabyte", "GV-N730-2GI", "GeForce GT 730", 700, "DirectX 11,
OpenGL 4.x", 16, 2, "DDR3", 800, 24447);

```

```

INSERT INTO `motherboard` (`mb_type`, `mb_socket`, `mb_chipset`,
`mb_maxRamMemory_gb`, `mb_bios`, `mb_ports`, `mb_img`, `mb_price_tg`)
VALUES
( "office", "Socket1155", "Intel H61", 8, "PnP 1.0a, DMI 2.0, SM BIOS 2.4,
ACPI 0b", "1x D-Sub / 1x PS/2 mouse / 1x PS/2 keyboard / 1x RJ45 / 3x audio jack
/ 4x USB-2.0/1.1", "img_path", 16135);

```

```
INSERT INTO `motherboard` (`mb_type`, `mb_socket`, `mb_chipset`,
`mb_maxRamMemory_gb`, `mb_bios`, `mb_ports`, `mb_img`, `mb_price_tg`)
VALUES
```

```
( "office-it", "Socket1151, MATX, iH110", "Intel H110", 32, "EFI AMI
DualBIOS", "1x PS/2 mouse port / 1x PS/2 Keyboard port / 1x D-Sub port / 2x
USB 3.0 ports / 2x USB 2.0 ports / 1x RJ-45 port / 3x audio jacks", "img_path",
20800);
```

```
INSERT INTO `motherboard` (`mb_type`, `mb_socket`, `mb_chipset`,
`mb_maxRamMemory_gb`, `mb_bios`, `mb_ports`, `mb_img`, `mb_price_tg`)
VALUES
```

Продолжение приложения Б

```
( "office-it", "Socket AM3+, MATX, AMD 760G", "AMD 760G / AMD
SB710", 16, "2x 16Mbit SM BIOS 2.4", "1x D-Sub / 1x DVI-D / 1x PS/2 mouse /
1x PS/2 keyboard / 1x RJ45 / 3x audio jack / 4x USB2.0/1.1", "img_path", 18000);
```

```
INSERT INTO `motherboard` (`mb_type`, `mb_socket`, `mb_chipset`,
`mb_maxRamMemory_gb`, `mb_bios`, `mb_ports`, `mb_img`, `mb_price_tg`)
VALUES
```

```
( "gaming", "Socket2011-3, EATX, iX99", "Intel X99 Express", 32, "UEFI
AMI, 2x 128Mbit DualBIOS", "2x PS/2 keyboard/mouse combo port / 4x USB 3.0
/ 4x USB 2.0 / 1x LAN (RJ-45) / 5x Audio jacks / 1x S/P-DIF", "img_path",
79000);
```

```
INSERT INTO `motherboard` (`mb_type`, `mb_socket`, `mb_chipset`,
`mb_maxRamMemory_gb`, `mb_bios`, `mb_ports`, `mb_img`, `mb_price_tg`)
VALUES
```

```
( "gaming", "Socket1151, ATX, iZ170", "Intel Z170", 64, "BIOS
Flashback+", "1x Clear CMOS button / 1x PS/2 keyboard/ mouse port / 3x USB 2.0
ports / 1x HOTKEY port) / 1x BIOS FLASHBACK+ port) / 1x DisplayPort / 1x
HDMI port / 1x USB 3.1 Gen2 port / 1x USB 3.1 Gen2 Type-C port / 1x LAN
(RJ45) port / 2x USB 3.1 Gen1 ports / 1x Optical S/PDIF OUT connector / 5x OFC
audio jacks", "img_path", 167000);
```

```
INSERT INTO `pc`(`cpu_id`, `coo_id`, `case_id`, `ps_id`, `hdd_id`,
`ram_id`, `vid_id`, `mb_id`) VALUES
```

```
(4, 6, 1, 2, 4, 1, 1, 1);
```

```
INSERT INTO `pc`(`cpu_id`, `coo_id`, `case_id`, `ps_id`, `hdd_id`,
`ram_id`, `vid_id`, `mb_id`) VALUES
```

```
(5, 6, 2, 2, 5, 1, 2, 3);
```

```
INSERT INTO `pc`(`cpu_id`, `coo_id`, `case_id`, `ps_id`, `ssd_id`, `ram_id`,  
`vid_id`, `mb_id`) VALUES  
(4, 11, 4, 2, 1, 1, 2, 1);
```

```
INSERT INTO `pc`(`cpu_id`, `coo_id`, `case_id`, `ps_id`, `hdd_id`, `ssd_id`,  
`ram_id`, `vid_id`, `mb_id`) VALUES  
(1, 2, 2, 5, 1, 1, 3, 6, 2);
```

```
INSERT INTO `pc`(`cpu_id`, `coo_id`, `case_id`, `ps_id`, `hdd_id`, `ssd_id`,  
`ram_id`, `vid_id`, `mb_id`) VALUES  
(2, 4, 4, 3, 4, 1, 3, 3, 3);
```

```
INSERT INTO `pc`(`cpu_id`, `coo_id`, `case_id`, `ps_id`, `hdd_id`,  
`ram_id`, `vid_id`, `mb_id`) VALUES  
(8, 5, 2, 3, 5, 4, 6, 2);
```

Продолжение приложения Б

```
INSERT INTO `pc`(`cpu_id`, `coo_id`, `case_id`, `ps_id`, `hdd_id`, `ssd_id`,  
`ram_id`, `vid_id`, `mb_id`) VALUES  
(7, 3, 6, 3, 1, 1, 4, 6, 4);
```

```
INSERT INTO `pc`(`cpu_id`, `coo_id`, `case_id`, `ps_id`, `hdd_id`, `ssd_id`,  
`ram_id`, `vid_id`, `mb_id`) VALUES  
(8, 10, 5, 6, 2, 2, 5, 8, 5);
```

```
INSERT INTO `pc`(`cpu_id`, `coo_id`, `case_id`, `ps_id`, `hdd_id`, `ssd_id`,  
`ram_id`, `vid_id`, `mb_id`) VALUES  
(10, 11, 3, 5, 5, 3, 6, 7, 5);
```