

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное учреждение
высшего образования
«Южно-Уральский государственный университет
(национальный исследовательский университет)»

Политехнический институт

Факультет механико-технологический
Базовая кафедра техники и технологии

ДОПУСТИТЬ К ЗАЩИТЕ

Заведующий кафедрой, к.т.н., до-
цент _____ А.В. Прохоров
_____ 2017 г.

Разработка автоматизированного рабочего места по контролю и учё-
ту компьютерной и оргтехники

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
К ВЫПУСКНОЙ КВАЛИФИКАЦИОННОЙ РАБОТЕ
ЮУрГУ– 090301.2017.452. ПЗ ВКР

Руководитель работы,
инженер

_____ А.В. Мехов
_____ 2017 г.

Автор работы –
студент группы ДО-580

_____ И.В. Лубинец
_____ 2017 г.

Нормоконтролер, старший пре-
подаватель

_____ Д.П. Химичева
_____ 2017 г.

Озерск 2017

АННОТАЦИЯ

Лубинец, И.В. Разработка автоматизированного рабочего места по контролю и учёту компьютерной и оргтехники. – Озерск: ЮУрГУ, ДО-580; 2017. – 62 с. 19 илл., 5 табл., библиогр. список – 49 наим., 2 прил., презентация на 16 слайдах.

В выпускной квалификационной работе разработано автоматизированное рабочее место по контролю и учёту компьютерной и оргтехники.

Предлагается: изучить теоретические основы проектирования баз данных, классификацию баз данных, а также виды компьютерных сетей, с помощью которых пользователь взаимодействует с базами данных. Спроектировать информационную структуру предметной области. Разработать физическую модель базы данных и пользовательский интерфейс доступа к данным. Разработать комплекс для учёта компьютерной и оргтехники.

					ЮУрГУ 090301.2017.452 ПЗ ВКР			
<i>Изм.</i>	<i>Лист</i>	<i>№ докум.</i>	<i>Подпись</i>	<i>Дата</i>				
<i>Разраб.</i>		Лубинец И.В.			Создание автоматизированного рабочего места по контролю и учёту компьютерной и оргтехники	<i>Лит.</i>	<i>Лист</i>	<i>Листов</i>
<i>Провер.</i>		Мехов А.В.				Д	2	62
<i>Реценз.</i>						ЮУрГУ базовая кафедра техники и технологии		
<i>Н. Контр.</i>		Химичева Д.П.						
<i>Утверд.</i>		Прохоров А.В.						

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ	4
1 АНАЛИЗ ТЕХНОЛОГИЙ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ НА ОСНОВЕ БАЗ ДАННЫХ	
1.1 Теоретические основы проектирования БД	6
1.2 Классификация архитектуры БД	9
1.3 Компьютерные сети и их виды	15
2 СРАВНЕНИЕ ОТЕЧЕСТВЕННЫХ И ПЕРЕДОВЫХ ЗАРУБЕЖНЫХ ТЕХНОЛОГИЙ И РЕШЕНИЙ	
2.1 Обзор программных продуктов для учета компьютерной техники и оргтехники	19
2.2 Обзор технологий и средств реализации для разработки автоматизированных систем	22
3 РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ УЧЕТА КОМПЬЮТЕРНОЙ И ОРГТЕХНИКИ	
3.1 Назначение программного комплекса	31
3.2 Проектирование информационной структуры предметной области.....	31
3.3 Программная реализация АРМ учета и контроля техники и оргтехники.....	35
3.4 Расчет технико-экономических показателей разработки программного комплекса	47
ЗАКЛЮЧЕНИЕ	55
ПЕРЕЧЕНЬ СОКРАЩЕНИЙ	56
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	57
ПРИЛОЖЕНИЯ	
ПРИЛОЖЕНИЕ А ЛОГИЧЕСКАЯ МОДЕЛЬ БАЗЫ ДАННЫХ	61
ПРИЛОЖЕНИЕ Б ФИЗИЧЕСКАЯ МОДЕЛЬ БАЗЫ ДАННЫХ	62

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		3

ВВЕДЕНИЕ

Актуальность темы. Автоматизация различных сфер деятельности предприятия, будь то сбор данных или управление технологическими процессами, позволяет ускорить работу, сделать ее более точной и эффективной, избежать потерь нужной информации, ошибок персонала, дублирования документов, запутанного порядка их прохождения. Для эффективности работы по учету техники необходимо проводить мероприятия по обновлению материально-технической и информационной базы, внедрению современных методов управления с использованием автоматизированных подсистем и автоматизированных рабочих мест работников, созданию банка данных о технике предприятия, его своевременному пополнению, оперативному представлению необходимой информации пользователям.

Целью выпускной квалификационной работы является разработка автоматизированной системы «Учет компьютерной техники и оргтехники».

Задачи выпускной квалификационной работы:

- изучить теоретические основы проектирования баз данных, классификацию баз данных, а также виды компьютерных сетей, с помощью которых пользователь взаимодействует с БД;
- изучить отечественные и зарубежные готовые коммерческие программные продукты по учету компьютерной техники, а также средства разработки автоматизированных систем;
- спроектировать информационную структуру предметной области;
- разработать физическую модель базы данных и пользовательский интерфейс доступа к данным;
- провести расчет технико-экономических показателей для разработанного программного комплекса.

Объектом выпускной квалификационной работы: процесс учета материальных активов в деятельности предприятия.

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		4

Предметом выпускной квалификационной работы является автоматизация учета компьютерной техники и оргтехники.

Практическая значимость выпускной квалификационной работы состоит в том, что использование средств автоматизации сокращает трудозатраты, требуемые для учета техники и оргтехники, за счет упрощения операции пополнения и представления информации по ремонтам, добавлениям, использованию и отказам, постоянного контроля за корректностью вводимой информации. Это сильно упрощает работу и исключает ошибки, часто встречающиеся при обычной организации работы.

Структура выпускной квалификационной работы состоит из введения, трех разделов, заключения и библиографического списка. Раздел 1 посвящен теоретическому обоснованию темы исследования, описывается классификация архитектур баз данных и современных компьютерных сетей. Раздел 2 посвящен анализу имеющихся российских и зарубежных программных аналогов для учета компьютерной техники на предприятии, а также средств реализации автоматизированных информационных систем. В разделе 3 дается описание этапов проектирования и разработки автоматизированной системы «Учет компьютерной техники и оргтехники», а также приводится расчет показателей экономической эффективности проекта.

Объем выпускной квалификационной работы составляет 62 страницы машинописного текста и содержит 19 иллюстраций, 5 таблиц, библиографический список из 52 наименований и 2 приложения.

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		5

1 АНАЛИЗ ТЕХНОЛОГИЙ ПРОЕКТИРОВАНИЯ ИНФОРМАЦИОННЫХ СИСТЕМ НА ОСНОВЕ БАЗ ДАННЫХ

1.1 Теоретические основы проектирования БД

Каждая информационная система в зависимости от ее назначения имеет дело с той или иной частью реального мира, которую называют предметной областью системы. Выявление предметной области – начальный этап разработки любой информационной системы [12].

В свою очередь предметная область рассматривается как некоторая совокупность реальных объектов или сущностей, каждая из которых обладает определенным набором свойств или атрибутов. Любой объект предметной области должен быть отличным от других объектов того типа, к которому он отнесен. С этой целью объектам данного типа назначается некоторый идентификатор или первичный ключ, позволяющий на них однозначно ссылаться. В качестве первичного ключа может использоваться какой-либо атрибут или комбинация нескольких атрибутов объекта. На практике часто используются и не уникальные идентификаторы – вторичные ключи, определяющие альтернативный порядок просмотра, и внешние ключи, служащие для организации связей между объектами.

При переходе от выявления предметной области к реализации автоматизированной информационной системы необходимо рассмотреть такие понятия как данные, БД, СУБД, модель данных.

Данные – это информация, зафиксированная в определенной структурированной форме, пригодной для последующей обработки, хранения и передачи. БД – это структурированная определенным образом совокупность данных, относящихся к конкретной предметной области. БД может быть как локальная, так и распределенная. СУБД представляет собой комплекс инструментальных средств (программных и языковых), реализующих централизованное управление БД и обеспечивающих доступ к данным (изменение, добавление, удаление, резервное копирование и т. д.). СУБД должна обеспечивать поиск, модификацию и сохранность данных, защиту целостности данных от аппаратных сбоев и программных

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		6

ошибок, а также разграничение прав пользователей и защиту от несанкционированного доступа [1].

При работе с СУБД можно выделить несколько уровней представления данных:

- внешний уровень (уровень конечного пользователя). В некотором смысле это самый главный уровень. Именно с ним работает конечный пользователь, который воспринимает данные как совокупность некоторых взаимосвязанных полей, позволяющих ему решать свою задачу;
- концептуальный уровень (уровень программиста и администратора) – это обобщенное представление обо всех данных, хранящихся в базе, или совокупность внешних представлений. На этом уровне работают программист, создающий прикладные программы, и администратор, разрабатывающий схему базы данных;
- физический уровень (уровень реализации). На физическом уровне определяются способы организации данных на носителе и методы доступа к ним. СУБД реализует именно этот уровень.

По логическому представлению модели данных делятся на три группы: иерархические, сетевые и реляционные [8].

Основной структурой в иерархических моделях данных является дерево, в вершинах которого располагаются типы записей. Каждая из вершин связана только с одной вершиной вышележащего уровня иерархии. Поиск данных в такой структуре выполняется всегда по одной из ветвей, начиная с корневого элемента, то есть должен быть указан полный путь движения по ветви.

Базовая структура сетевых моделей – граф общего вида. По сравнению с иерархической моделью данных никаких ограничений на количество связей, входящих в каждую вершину, не накладываемся, что позволяет отображать связи между объектами предметной области практически любой степени сложности. При таком представлении существует несколько входов в сеть – неоднозначность доступа к данным.

Недостатки рассмотренных моделей:

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		7

- достаточно сложные механизмы доступа к данным, особенно в сетевой модели;
- основной единицей обработки в сетевой и иерархической моделях является запись.

Реляционная модель строится на использовании табличных методов и средств представления данных и манипулирования ими. Связь между таблицами может находить свое отражение в структуре данных, а может только подразумеваться, то есть присутствовать на неформальном уровне. Каждая таблица БД представляется как совокупность строк и столбцов, где строки соответствуют экземпляру объекта, а столбцы – атрибутам. В терминологии теории реляционных БД таблицам соответствуют отношения, столбцам – атрибуты, строкам – кортежи. Отношениям и атрибутам присваиваются имена. При практической разработке БД таблицы так и называются таблицами, строки – записями, столбцы – полями [10].

К отношениям реляционной модели данных предъявляется ряд требований:

- значения атрибутов являются атомарными;
- в отношении не может быть двух одинаковых кортежей;
- порядок следования атрибутов фиксирован, но так как атрибуты имеют имя, порядок не играет существенной роли;
- порядок следования кортежей безразличен.

Между двумя и более таблицами базы данных могут существовать отношения подчиненности. Отношения подчиненности определяют, что для каждой записи главной таблицы (master, называемой еще родительской) может существовать одна или несколько записей в подчиненной таблице (detail, называемой еще дочерней). Существует две разновидности связей между таблицами базы данных: «один-ко-многим», «многие-ко-многим» [8].

Кроме того, реляционные БД имеют мощный теоретический фундамент, основанный на математической теории отношений, и специальный язык структурированных запросов SQL, ориентированный на высокоуровневые операции с данными.

					090301.2017.452 ПЗ	Лист
						8
Изм.	Лист	№ докум.	Подпись	Дата		

Следует отметить, что реляционные БД практически вытеснили БД других видов. Основной причиной этого явилась сложность представления данных в иерархической и сетевой моделях и необходимость определения связей между данными на этапе проектирования БД, в то время как в реляционных БД связи между таблицами могут устанавливаться непосредственно в момент выполнения запросов. К тому же, разработчикам и пользователям значительно проще отображать сущности предметной области в табличных структурах данных [1].

1.2 Классификация архитектуры БД

БД на персональных компьютерах развивались по направлению от настольных (desktop), или локальных приложений, когда реально с БД могло работать одно приложение, до систем коллективного доступа к БД [10]. Имеется четыре разновидности архитектур баз данных:

- локальные базы данных;
- архитектура «файл-сервер»;
- архитектура «клиент-сервер»;
- распределенная архитектура.

Использование той или иной архитектуры накладывает сильный отпечаток на общую идеологию работы приложения, на программный код в приложении. Ниже представлено краткое описание перечисленных разновидностей архитектур баз данных.

Локальные БД и архитектура «файл-сервер».

При работе с локальными базами данных сами БД расположены на том же компьютере, что и приложения, осуществляющие доступ к ним. Работа с БД происходит в однопользовательском режиме. Приложение ответственно за поддержание целостности БД и за выполнение запросов к БД [15].

При работе в архитектуре «файл-сервер» БД и приложение расположены на файловом сервере сети (например, Novell NetWare). Возможна многопользовательская работа с одной и той же БД, когда каждый пользователь со своего компьютера запускает приложение, расположенное на сетевом сервере. Тогда на компьютере пользователя запускается копия приложения. По каждому запросу к

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		9

БД из приложения данные из таблиц БД перегоняются на компьютер пользователя, независимо от того, сколько реально нужно данных для выполнения запроса. После этого выполняется запрос.

Каждый пользователь имеет на своем компьютере локальную копию данных, время от времени обновляемых из реальной БД, расположенной на сетевом сервере. При этом изменения, которые каждый пользователь вносит в БД, могут быть до определенного момента неизвестны другим пользователям, что делает актуальной задачу систематического обновления данных на компьютере пользователя из реальной БД. Другой актуальной задачей является блокирование записей, которые изменяются одним из пользователей; это необходимо для того, чтобы в это время другой пользователь не внес изменений в те же данные.

В архитектуре «файл-сервер» вся тяжесть выполнения запросов к БД и управления целостностью БД ложится на приложение пользователя. БД на сервере является пассивным источником данных. Общая схема архитектуры «файл-сервер» показана на рисунке 1. Кардинальных различий с точки зрения архитектуры между однопользовательской архитектурой и архитектурой «файл-сервер» нет. И в том, и в другом случае в качестве СУБД применяются так называемые «персональные» (или «локальные») СУБД, такие как Paradox, dBase, FoxPro и пр. Сама база данных в этом случае представляет собой набор таблиц, индексных файлов, файлов полей комментариев (мето-полей) и пр., хранящихся в одном каталоге на диске в виде отдельных файлов.

					090301.2017.452 ПЗ	Лист
						10
Изм.	Лист	№ докум.	Подпись	Дата		

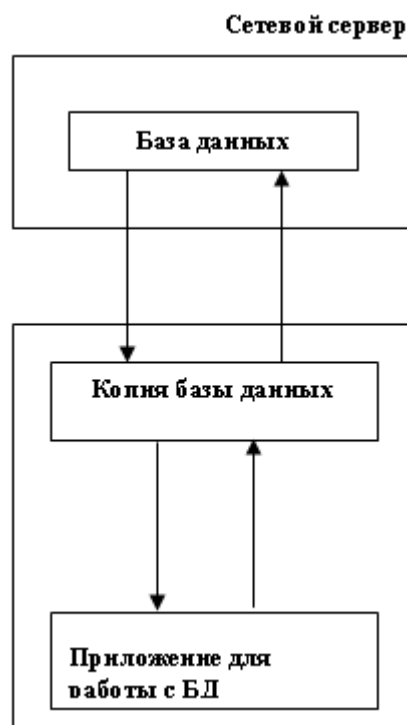


Рисунок 1 – Архитектура «Файл-сервер»

Удаленные БД и архитектура «клиент-сервер».

Архитектура «файл-сервер» неэффективна, по крайней мере, в двух отношениях:

- При выполнении запроса к базе данных, расположенной на файловом сервере, в действительности происходит запрос к локальной копии данных на компьютере пользователя. Поэтому перед выполнением запроса данные в локальной копии обновляются в полном объеме. Так, если таблица БД состоит из 1000 записей, а для выполнения запроса (например, выдать сумму премий за октябрь в отделе Y) реально нужно 10 записей, все равно перегоняются все 1000 записей. Таким образом, не нужно иметь слишком много пользователей и запросов от них, чтобы серьезно «забить» сеть, что, конечно же, не может не сказаться на ее скорости.

- Обеспечение целостности БД производится из приложений. Это потенциальный источник ошибок, нарушающих физическую и логическую целостность БД, поскольку различные приложения могут производить контроль целостности по-разному, взаимоисключающими способами, или не производить такого контроля вовсе. Намного эффективнее управлять БД из единого места и по единым

законам (все зависит от того, как написано приложение). Поэтому безопасность при работе в архитектуре «файл-сервер» невысока и всегда присутствует элемент неопределенности. Секретность и конфиденциальность при работе с БД в архитектуре «файл-сервер» обеспечить также тяжело – любой, кто имеет доступ в каталог сетевого сервера, где хранится БД, может изменять таблицы любым образом, копировать их, заменять и т. д.

Архитектура «клиент-сервер» (рисунок 2), разделяет функции приложения пользователя (называемого клиентом) и сервера [4]. Приложение-клиент формирует запрос к серверу, на котором расположена БД, на специальном структурном языке запросов SQL, являющемуся промышленным стандартом в мире реляционных БД [23]. Удаленный сервер, называемый сервером приложений, принимает запрос и переадресует его SQL-серверу БД.

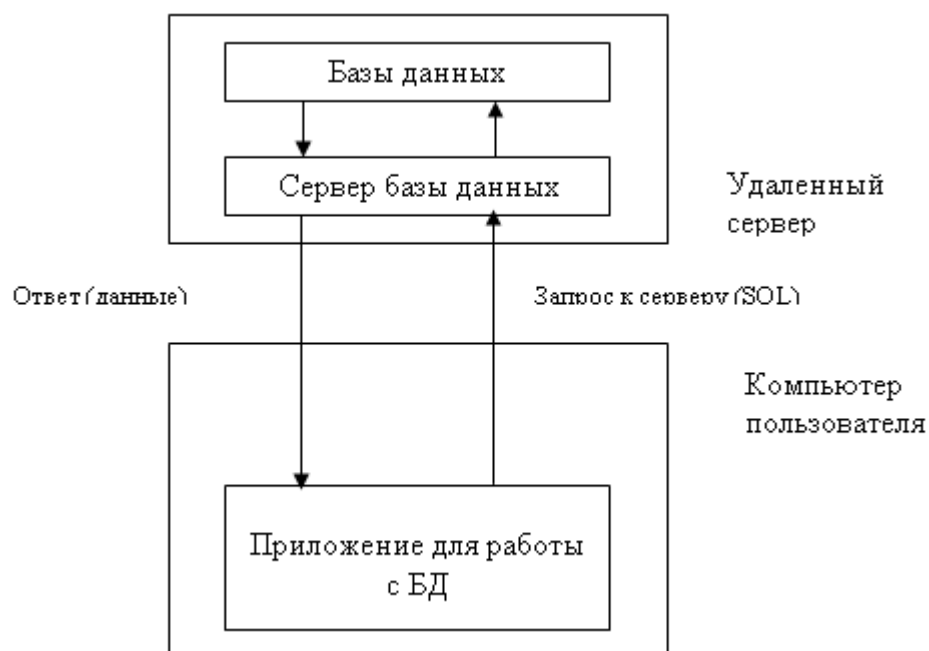


Рисунок 2 – Архитектура «Клиент-сервер»

SQL-сервер – специальная программа, управляющая удаленной БД. SQL-сервер обеспечивает интерпретацию запроса, его выполнение в БД, формирование результата выполнения запроса и выдачу его приложению-клиенту. При этом ресурсы клиентского компьютера не участвуют в физическом выполнении запроса; клиентский компьютер лишь отправляет запрос к серверной БД и получает результат, после чего интерпретирует его необходимым образом и представляет пользователю.

- Хранение бизнес-правил, часто используемых запросов в уже интерпретированном виде.
- Обеспечение одновременной, безопасной и отказоустойчивой многопользовательской работы с одними и теми же данными.

В архитектуре «клиент-сервер» используются так называемые «удаленные» (или «промышленные») СУБД [4]. Промышленными они называются потому, что именно СУБД этого класса могут обеспечить работу информационных систем масштаба среднего и крупного предприятия, организации, банка. Локальные СУБД предназначены для однопользовательской работы или для обеспечения работы информационных систем, рассчитанных на небольшие группы пользователей. К разряду промышленных СУБД принадлежат Oracle, Gupta, Informix, Sybase, MS SQL Server, DB2, InterBase и ряд других.

Как правило, SQL-сервер управляется отдельным сотрудником или группой сотрудников (администраторы SQL-сервера). Они управляют физическими характеристиками баз данных, производят оптимизацию, настройку и переопределение различных компонентов БД, создают новые БД, изменяют существующие и т. д., а также выдают привилегии (разрешения на доступ определенного уровня к конкретным БД, SQL-серверу) различным пользователям. Кроме этого, существует отдельная категория сотрудников, называемых администраторами баз данных. Как правило, это администраторы сервера, разработчики БД или пользователи, имеющие привилегии на создание, изменение, настройку оптимальных параметров отдельных серверных БД. Администраторы БД также отвечают за предоставление прав на разноуровневый доступ к сопровождаемым ими БД для других пользователей.

Использование архитектуры «клиент-сервер»:

- резко уменьшает сетевой трафик;
- понижает сложность приложений-клиентов (поскольку тем уже нет необходимости обеспечивать целостность и безопасность БД и следить за параметрами многопользовательской работы с БД);

										Лист
										14
Изм.	Лист	№ докум.	Подпись	Дата	090301.2017.452 ПЗ					

название сетевого взаимодействия. Взаимодействие сетевых компонентов – синергетика сети – воплощает старую идею о том, что сумма больше чем совокупность слагаемых: компоненты эффективнее работают вместе и реализуют более мощные возможности, если их рассматривать независимо [13].

Основное различие между сетью и любой другой компьютерной системой определяется местом, в котором происходит обработка. В системах с большой или мини-ЭВМ всю работу выполняет центральная машина. «Подчиненные» клиенты работают просто как неинтеллектуальные терминалы. Сеть же отличается тем, что работу в ней выполняют клиенты. Например, в локальной сети NetWare все данные хранятся на центральном файловом сервере. Когда пользователь хочет получить доступ к данным, он подключается к файловому серверу и загружает копию данных в оперативную память локальной рабочей станции, которая выполняет приложения и работает с сетевыми данными – все это происходит в ОЗУ данной станции. Такая обработка называется распределенной [13]. Когда пользователь завершает операции с данными, он передает их обратно на файловый сервер, где они хранятся до следующего использования. Наиболее впечатляющим в распределенной обработке является то, что каждый пользователь имеет доступ ко всей компьютерной системе. В централизованной системе клиентские терминалы используют часть мощности центрального процессора. В распределенной системе при подключении дополнительных рабочих станций ее общая вычислительная мощность возрастает. Преимущество распределенной обработки состоит также в повышенной защите, параллельной обработке данных и отказоустойчивости системы.

Первоначально компьютерные сети были небольшими и объединяли до десяти компьютеров и один принтер. Технология ограничивала размеры сети, в том числе количество компьютеров в сети и ее физическую длину. В начале 1980-х годов самый популярный тип сетей состоял не более чем из 30 компьютеров, а длина ее кабеля не превышала 185 м (600 футов). Такие сети располагались в пределах одного этажа здания или небольшой организации. Для маленьких фирм эта конфигурация подходит и сегодня. Эти сети называются локальными

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		16

вычислительными сетями (ЛВС, LAN) [13]. Один из наиболее привлекательных аспектов таких сетей – скорость коммуникаций.

Первые типы локальных сетей не могли удовлетворить потребности крупных предприятий, офисы которых обычно расположены в различных местах. Но как только преимущества компьютерных сетей стали неоспоримы и все больше сетевых программ появлялось на рынке, перед корпорациями – для сохранения конкурентоспособности – встала задача расширения сетей. Так на основе локальных сетей возникли более крупные системы – глобальные вычислительные сети (ГВС, WAN). Эти системы объединяют в себе несколько локальных сетей. Можно сказать, что это своего рода сеть из локальных сетей. В глобальной сети принципы локальных сетей распространяются на более протяженные географические пространства и более широкие технические области. Такое географическое расширение достигается с помощью международных коммуникационных каналов, например каналов телефонной связи. Технического расширения удается добиться за счет интеграции различных машин, «говорящих на разных языках». За такую расширенную стыкуемость и связь приходится расплачиваться производительностью. Глобальные сети не обеспечивают быстродействия локальных сетей. Все эти виды сетей объединяет нечто общее – связь, являющаяся основой всех вычислительных сетей. Их цель состоит в совместном использовании ресурсов с помощью взаимосвязанных каналов, а также в обеспечении постоянной связи в реальном режиме времени. Понятие связи в реальном режиме времени подразумевает постоянный обмен сообщениями. Ресурсы – это данные, приложения и периферийные устройства, такие, как принтер, модем или сканер. Сетевая связь достигается с помощью одноранговых коммуникаций, коммуникаций типа клиент-сервер или межкомпонентного соединения [13].

Таким образом, компьютерные сети позволяют использовать ресурсы, например файлы и принтеры, а также работать с интерактивными приложениями, например планировщиками и электронной почтой. Компьютерные сети обеспечивают множество преимуществ. Их использование, в частности, позволяет:

- снизить затраты, благодаря совместному использованию данных и периферийных устройств;

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		17

- унифицировать приложения;
- своевременно получать данные;
- эффективно взаимодействовать с партнерами, гибко планировать свое рабочее время.

Сети на основе технологий «клиент-сервер», «файл-сервер» – это наиболее эффективный способ обеспечить:

- Доступ к базам данных таких приложений, как:
 - электронные таблицы;
 - бухгалтерские программы;
 - коммуникационные приложения;
 - системы управления документами.
- Централизованное хранение файлов.

Применение «клиент-сервер» необходимо для организации доступа к данным, если требуется разгрузить клиентское приложение и снизить трафик за счет обработки запросов сервером. В случае незначительного количества данных и количества клиентов, которые могут использовать одновременно БД, возможно применение технологии «файл-сервер».

Применение технологии БД целесообразно в любой крупной организации, где значительному числу сотрудников необходим постоянный доступ к данным большого объема.

С возрастанием потребностей в оперативности получения информации и высоким темпом развития вычислительной техники и сетевых технологий растет потребность усовершенствования программного обеспечения, автоматизирующих работу предприятий.

					090301.2017.452 ПЗ	Лист
						18
Изм.	Лист	№ докум.	Подпись	Дата		

2 СРАВНЕНИЕ ОТЕЧЕСТВЕННЫХ И ПЕРЕДОВЫХ ЗАРУБЕЖНЫХ ТЕХНОЛОГИЙ И РЕШЕНИЙ

2.1 Обзор программных продуктов для учета компьютерной техники и оргтехники

В эпоху повсеместной компьютеризации практически любое предприятие сталкивается с необходимостью ведения учета используемой в производственной или бизнес-деятельности компьютерной техники, в том числе и различной оргтехники и комплектующих. И чем больше масштаб деятельности предприятия, тем сложнее вести подобный учет. Для решения данной проблемы можно воспользоваться готовым коммерческим программным продуктом, либо разработать уникальную информационную систему под требования конкретного предприятия.

На сегодняшний момент на рынке информационных технологий предлагается целый ряд программных продуктов, которые решают проблему эффективного учета компьютерной и оргтехники. Ниже представлено краткое описание наиболее популярных из них.

Среди отечественных производителей ИТ-продуктов наиболее востребованными являются программные приложения, в основе которых лежит платформа 1С отечественного разработчика «Фирма 1С». Продукты данной компании охватывают практически все сферы бизнеса, подлежащие автоматизации.

Для решения задач автоматизации сферы ИТ ими был разработан специальный программный продукт – «1С:ИТIL Управление информационными технологиями предприятия», который представляет собой универсальное решение для организации комплексного управления процессами в сфере информационных технологий.

Основным назначением «1С:ИТIL» является автоматизация следующих ИТ-процессов и функций:

- управление каталогом и уровнем услуг;
- управление конфигурациями и активами;

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		19

- выполнение запросов;
- управление инцидентами;
- управление изменениями;
- управление проблемами;
- управление релизами;
- управление событиями;
- Service Desk.

Рассматриваемая в данной работе функция учета компьютерной и оргтехники реализуется с помощью подсистемы «Управление конфигурациями и активами», которая позволяет вести эффективный учет материальных и нематериальных ИТ-ресурсов, а именно:

- учет компьютерной и оргтехники в разрезе количества и суммы, серийных номеров и штрих-кодов, а также учет ИТ-активов в разрезе дополнительных свойств и характеристик (например, производитель, напряжение питания, тип, размер и т. д.);
- отслеживание окончания гарантийных сроков оборудования;
- контроль выполнения технического обслуживания и т. д.

То есть учет компьютерной техники является лишь одной из многочисленных функций данного программного решения, поэтому использование «1С:ИТIL» целесообразно только на крупных предприятиях, успешная бизнес-деятельность которых в значительной степени зависит от слаженного и бесперебойного функционирования всей ИТ-инфраструктуры, когда затраты на внедрение и техническую поддержку программного продукта окупятся в минимальные сроки. Для малых и средних предприятий функционал, предоставляемый данным программным решением, во многом является избыточным, а затраты – практически не окупаемыми.

Еще одним примером успешной реализации программного продукта по учету компьютерной техники является ПО «IT Invent» отечественного производства. Данная программа позволяет вести инвентаризационный учет

					090301.2017.452 ПЗ	Лист
						20
Изм.	Лист	№ докум.	Подпись	Дата		

компьютеров, оргтехники, периферийного и ИТ-оборудования, программного обеспечения, комплектующих и расходных материалов.

В качестве ключевых особенностей программы можно отметить следующее:

- поддержка базы данных MS Access и MS SQL Server;
- многопользовательский режим работы;
- создание и настройки собственных дополнительных свойств различных типов;
- учет заказов поставщикам на все виды учетных единиц;
- учет выполнения любых работ внутри организации, в том числе учет ремонтов и профилактических обслуживаний оборудования и компьютеров;
- уникальная система создания и печати инвентарных этикеток;
- поддержка работы со сканером штрих-кодов поиск записей по штрих-коду;
- ведение истории изменений по оборудованию;
- учет расходных материалов, комплектующих запчастей и канцелярии;
- ведение базы поставщиков, сервисных организаций и прочих контрагентов.
- большое количество встроенных печатных форм и отчетов с возможностью их редактирования и др.

Данный продукт распространяется также на коммерческой основе. при этом стоимость лицензии зависит от количества учетных единиц в базе данных. В случае, когда предприятие активно развивается, такое ограничение может привести к ситуации, когда раз за разом придется докупать право на ведение учета большего количества оборудования, что в конечном итоге может привести к значительным и необоснованным затратам.

Кроме этого, рассмотренные программные решения обладают довольно сложным и перегруженным пользовательским интерфейсом, ориентированным по большей части на «продвинутых» пользователей, что может потребовать дополнительных затрат на обучение персонала по работе с данными программными системами.

					090301.2017.452 ПЗ	Лист
						21
Изм.	Лист	№ докум.	Подпись	Дата		

Таким образом, с точки зрения реализации задач данной выпускной квалификационной работы, разработка нового программного продукта для целей учета компьютерной и оргтехники, который будет выполнять строго ограниченный набор функций, является наиболее предпочтительной. При этом, разработанное программное решение всегда можно будет дополнить функционалом по требованию заказчика.

2.2 Обзор технологий и средств реализации для разработки автоматизированных систем

На современном этапе развития компьютерных технологий для создания автоматизированных информационных систем существует множество вариантов выбора СУБД, языка программирования и среды разработки. Выбор технологий и средств реализации в каждом конкретном случае определяется из следующих основных параметров: требования к аппаратной и программной совместимости, объем и характер обрабатываемых данных, желания заказчика и предпочтения разработчика.

Поскольку разрабатываемая программная система по учету компьютерной техники и оргтехники будет строиться на основе архитектуры «клиент-сервер» с использованием технологий баз данных, то в первую очередь необходимо определиться именно с системой управления базой данных.

На данный момент имеется множество СУБД, которые можно с успехом использовать для реализации учета компьютерной и периферийной техники на предприятии. Однако лидерами по популярности являются все же СУБД зарубежного производства (MS Access, MS SQL Server, MySQL и т. д.), российские аналоги, к сожалению, пока далеки от повсеместного использования. Рассмотрим кратко основные особенности перечисленных СУБД.

Microsoft Access (MS Access) – система управления базами данных, предназначенная для создания и обслуживания баз данных, обеспечения доступа к данным и их обработки. MS Access предоставляет максимальную свободу в задании типа данных (текст, числовые данные, даты, время, денежные значения, рисунки, электронные таблицы), в том числе позволяет задавать форматы

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		22

хранения предоставления этих данных при выводе на экран или печать. При этом СУБД MS Access позволяет размещать в своих структурах не только данные, но и методы (программные коды), что предоставляет широкие возможности по использованию средств для хранения информации и эффективного управления этой информацией.

MS Access – это функционально полная реляционная СУБД. В ней предусмотрены все необходимые разработчику средства для определения и обработки данных, а также для управления ими при работе с большими объемами информации.

В MS Access для обработки данных базовых таблиц используется мощный язык SQL. Используя SQL, можно выделить из одной или нескольких таблиц необходимую для решения конкретной задачи информацию. СУБД MS Access спроектирована таким образом, что она может быть использована как в качестве самостоятельной СУБД на отдельной рабочей станции, так и в сети – в режиме «клиент-сервер». Поскольку в MS Access к данным могут иметь доступ одновременно несколько пользователей, в нем предусмотрены надежные средства защиты и обеспечения целостности данных.

MS Access входит в самый популярный и востребованный во всем мире пакет офисных программ MS Office Professional и обладает простым и понятным интерфейсом, что позволяет его использовать при разработке как крупных многофункциональных информационных систем, так и для небольших локальных приложений.

Производителем СУБД SQL Server также является корпорация Microsoft, однако, в отличие от MS Access, данная СУБД ориентирована на профессиональных разработчиков, обладает расширенным функционалом, обеспечивает более высокую защиту данных, и, как правило, используется для построения корпоративных многопользовательских распределенных систем.

Основными задачами СУБД SQL Server являются организация одновременного доступа к данным большого количества пользователей, а также манипуляция информацией, хранимой в БД. SQL Server поддерживает реляционную модель данных.

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		23

Рассмотрим кратко основные принципы взаимодействия в СУБД SQL Server:

- SQL Server может устанавливаться на платформах Windows NT;
- SQL Server выполняет функции создания объектов базы данных (таблиц, индексов и т. д.), осуществляет проверку целостности базы данных и отвечает за безопасность данных в системе;
- доступ пользователя к данным СУБД SQL Server обычно осуществляется с компьютера рабочей станции. При этом создаются соответствующие приложения, которые позволяют выполнять операции над данными;
- для выполнения ряда задач, связанных в основном с администрированием базы данных системы SQL Server, обращение к системе удобно выполнять непосредственно с самого компьютера-сервера;
- в СУБД SQL Server для манипулирования данными используется язык запросов Transact-SQL, который является переработанной компанией Microsoft версией языка SQL.

СУБД SQL Server сохраняет создаваемые объекты в соответствующих файлах на диске компьютера-сервера. При этом для таких объектов, как база данных, создаются специальные таблицы. В них хранится информация о различных элементах базы данных: индексах, таблицах, пользователях и т. д. Файлы базы данных сохраняются с расширением *.MDF, а системные файлы с расширением *.LDF.

Работая на более интеллектуальных системах в качестве клиентов, пользователи могут получать информацию и манипулировать ею локально. Архитектура этого типа предполагает возможность оптимизации обработки информации, позволяя каждому компоненту работать с данными независимо и наиболее подходящим для конкретного компонента способом. Сервер занимается только обработкой информации в БД, а клиент – представлением этой информации.

При доступе клиента к данным выходят на первый план две ключевые характеристики БД сервера. Первая обеспечивает единую точку доступа к БД

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		24

сервера. Вторая разделяет обработку и манипуляцию данными между клиентом и сервером.

SQL Server позволяет приложениям клиента контролировать информацию, получаемую с сервера, с помощью различных специальных инструментов и технологий. Сюда включаются такие средства, как хранимые процедуры, поддерживаемые сервером правила, дающие возможность автоматической обработки данных.

Не менее популярной среди профессиональных разработчиков является и СУБД MySQL. СУБД MySQL – это полнофункциональная, свободно распространяемая система управления реляционными базами данных. MySQL начали разрабатывать в 1990-х годах, поскольку потребность в разумном управлении компьютерной информацией постоянно росла. Разработчики ядра MySQL пытались решить эту проблему с помощью маленькой и простой базы данных mSQL. Когда выяснилось, что mSQL не справляется со всеми задачами, которые создателям хотелось на нее возложить, они создали более мощную базу данных, которая превратилась в MySQL.

MySQL поддерживает несколько различных механизмов базы данных (database engines). Механизмы базы данных определяют, как MySQL в данный момент обрабатывает хранение и извлечение данных. Как следствие, каждый механизм хранения обладает собственным набором возможностей и преимуществ. Со временем имеющиеся механизмы базы данных становятся все более мощными и быстрыми.

Действующей версией MySQL считается последняя доступная версия 5.x. MySQL 5.x по производительности сопоставима с любой из гораздо более дорогих баз данных уровня предприятия, например Oracle, Informix, DB2 (IBM) или SQL Server (Microsoft). Такое повышение производительности стало возможным благодаря усилиям многих талантливых разработчиков открытого исходного кода, а также тестированию в сообществе. Однако СУБД MySQL все же наиболее часто используется для создания веб-ориентированных информационных систем в связке с веб-сервером и языком PHP.

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		25

Любая из рассмотренных СУБД может быть с успехом использована для разработки АРМ учета компьютерной техники и оргтехники, и в контексте задач данного дипломного проекта выбор той или иной СУБД зависит от предпочтений разработчика и его знаний и опыта, на основании которых в качестве СУБД для разрабатываемого программного решения по учету компьютерной техники была выбрана СУБД SQL Server.

Не менее важным для успешной реализации проекта является выбор среды программирования. В данном обзоре будут рассмотрены среды разработки Visual Studio и Delphi.

Microsoft Visual Studio – линейка продуктов компании Microsoft, включающих интегрированную среду разработки программного обеспечения и ряд других инструментальных средств. Данные продукты позволяют разрабатывать как консольные приложения, так и приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms, а также веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Microsoft Silverlight.

Visual Studio позволяет совместно использовать различные ресурсы и упрощает создание решений на базе нескольких языков: Visual Basic, Visual C# и Visual C++. Кроме того, в этих языках используются функциональные возможности платформы .NET Framework, которая позволяет получить доступ к ключевым технологиям, упрощающим разработку веб-приложений ASP и XML (веб-службы).

Visual Studio включает в себя редактор исходного кода с поддержкой технологии IntelliSense и возможностью простейшего рефакторинга кода. Встроенный отладчик может работать как отладчик уровня исходного кода, так и как отладчик машинного уровня. Остальные встраиваемые инструменты включают в себя редактор форм для упрощения создания графического интерфейса приложения, веб-редактор, дизайнер классов и дизайнер схемы базы данных и др.

В отличие от большинства других приложений и сред разработки, которые решают задачи чтения и обновления информации в базе данных, Visual Studio

					090301.2017.452 ПЗ	Лист
						26
Изм.	Лист	№ докум.	Подпись	Дата		

предоставляет поддержку для интеграции данных из БД непосредственно в приложение. Для этого Visual Studio предлагает широкий набор визуальных инструментов для работы с базами данных, которые помогают создавать и обслуживать базы данных, а также управлять приложениями, работающими с данными. Визуальные инструменты для баз данных позволяют подключиться к базе данных, просматривать ее структуру и данные, а также изменять их.

Не менее мощной по своему функционалу является среда разработки Borland Delphi. Delphi – это потомок среды программирования Turbo Pascal. Система визуального объектно-ориентированного проектирования Delphi позволяет:

- создавать законченные приложения для Windows самой различной направленности;
- быстро создавать профессионально выглядящий оконный интерфейс для любых приложений; интерфейс удовлетворяет всем требованиям Windows и автоматически настраивается на ту систему, которая установлена, поскольку использует функции, процедуры и библиотеки Windows;
- создавать свои динамически присоединяемые библиотеки компонентов, форм, функций, которые потом можно использовать из других языков программирования;
- создавать мощные системы работы с базами данных любых типов;
- формировать и печатать сложные отчеты, включающие таблицы, графики и т. п.;
- создавать справочные системы, как для своих приложений, так и для любых других;
- создавать профессиональные программы установки для приложений Windows, учитывающие всю специфику и все требования операционной системы.

Широкая поддержка БД – одно из главных свойств среды программирования Delphi. В настоящее время существует два наиболее распространенных способа хранения БД. Первый, состоит в том, что вся БД представляется как единый файл; вторая, каждая таблица, индекс и другие элементы БД располагаются в отдельных файлах. Delphi можно настроить на

					090301.2017.452 ПЗ	Лист
						27
Изм.	Лист	№ докум.	Подпись	Дата		

работу с обеими основными структурами хранения БД. На базу данных всегда ссылаются по ее имени или псевдониму, но эта ссылка может относиться как к файлу БД, так и к каталогу, содержащему файлы с таблицами. Delphi не привязана к определенному формату данных.

Приложения БД разработанные в Delphi не имеют прямого доступа к источнику данных, с которыми они работают. Delphi взаимодействует с Borland Database Engine (BDE), который непосредственно обращается к ряду источников данных, включая такие распространенные форматы как dBase, Paradox и таблицы ASCII. BDE также может взаимодействовать с Borland SQL Links, который позволяет обращаться к ряду локальных или удаленных серверов SQL. Из локальных серверов доступен InterBase для Windows; удаленные серверы включают Oracle, Sybase, Informix, InterBase, SQL Server. Если необходимо обращение к другим базам или форматам данных, то BDE может взаимодействовать с драйверами ODBC.

Кроме того, при помощи технологии OLE (Object Linking and Embedding) осуществляется взаимодействие с MS Excel, и MS Word и другими приложениями. Причем технология OLE позволяет не только взаимодействовать с приложениями, но и программно управлять ими, вызывая их методы через интерфейс OLE. Для возможности обращения к серверам автоматизации в Delphi создана панель Services (Службы), на которой расположены готовые компоненты (например, WordApplication, WordDocument), позволяющие быстро и легко обращаться к наиболее популярным офисным приложениям Microsoft, а также к созданным с их помощью документам.

Выбор среды разработки и программирования также во многом зависит от предпочтений разработчика, поскольку конечный пользователь не имеет доступа к исходному коду программы, ему важны удобство использования и стабильная работа приложения. Исходя из предпочтений и знаний разработчика, в качестве среды разработки была выбрана среда программирования Delphi.

Что касается совместимости аппаратного и программного обеспечения, то в отсутствие специализированных требований к аппаратному взаимодействию

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		28

компонентов разрабатываемого программного обеспечения, реализация требований совместимости сводится к выбору операционной системы.

Разрабатываемый программный комплекс и СУБД работают под управлением ОС семейства Microsoft Windows XP/2000/2003, а именно: для серверной части – Microsoft Windows Server 2003, а для клиентской – Microsoft Windows 2000 Professional или Microsoft Windows XP. Microsoft Windows – мощная, быстродействующая, переносимая, с поддержкой различных сетевых архитектур и различных процессоров ОС.

ОС этого семейства обеспечивает пользователей единым графическим интерфейсом, что сокращает время на подготовку пользователей и разработку графических интерфейсов прикладных программ. Работа в Windows основана на интуитивно понятных принципах, т. е. легко переключаться с задачи на задачу и осуществлять обмен информацией между ними. А также Windows обеспечивает переключение задач, управление виртуальной памятью, дает возможность перемещения объектов (drag-and-drop) и использует стандартные соглашения для обычных операций.

Вот некоторые характеристики ОС семейства Microsoft Windows [19]:

1. Встроенные сетевые возможности: ОС предназначена для работы в сети; поддерживает множество сетевых архитектур.
2. Файловые системы: Windows NT поддерживает три файловые системы FAT, NTFS, CDFS. Microsoft Windows поддерживает FAT и VFAT (улучшенная FAT с поддержкой длинных имен файлов).
3. Поддержка виртуальной памяти: виртуальная память позволяет ОС этого семейства одновременно выполнять несколько приложений на машинах с малым объемом оперативной памяти.
4. Надежность: приложения Windows выполняются в отдельных адресных пространствах. Аварийные приложения не влияют на работу других. Свойства архитектуры Windows защищают ОС от приложений, пытающихся занять слишком много процессорного времени или использовать адресное пространство ОС.

					090301.2017.452 ПЗ	Лист
						29
Изм.	Лист	№ докум.	Подпись	Дата		

С точки зрения разработчика программного обеспечения написание программ для Windows основано на использовании платформы Win32 API. Win32 API – название интерфейса прикладных программ, т. е. совокупности функций, к которым может обращаться приложение. Эта платформа поддерживается всеми ОС рассматриваемого семейства. Это означает, что тексты программ не придется переписывать для каждой ОС заново [6].

Windows – переносимая система, это означает, что она способна работать на машинах с разными типами процессоров. Все эти преимущества сделали ОС Windows самой популярной и распространенной системой, для которой разрабатывается все большее количество прикладных программ.

Таким образом, архитектура «клиент-сервер», в основе которой лежит использование СУБД SQL Server и среды разработки Delphi, наилучшим образом подходит для разработки современного программного обеспечения и для достижения целей данной выпускной квалификационной работы.

					090301.2017.452 ПЗ	Лист
						30
Изм.	Лист	№ докум.	Подпись	Дата		

3 РАЗРАБОТКА ПРОГРАММНОГО КОМПЛЕКСА ДЛЯ УЧЕТА КОМПЬЮТЕРНОЙ И ОРГТЕХНИКИ

3.1 Назначение программного комплекса

Данный программный комплекс рассматривается как один из этапов разработки автоматизированной системы «Учет компьютерной техники и оргтехники», поэтому он будет решать лишь следующие задачи:

- ведение справочников техники, используемой на предприятии;
- учет по местам хранения и установки;
- ведение справочника по ремонту;
- получение сводных данных о комплектующих, используемых в компьютере;
- подготовка данных для проведения инвентаризаций.

Создание этого продукта преследует следующие цели:

- автоматизация и совершенствование учета и управления;
- снижение трудоемкости и упорядочение деятельности по управлению.

При этом необходимо учитывать следующие требования:

- БД должна быть способна хранить информацию о 700 учетных единицах техники;
- стиль интерфейса пользователя: диалоговые окна в виде графических форм изменяемого размера с удобной навигацией по ней с помощью мыши.

3.2 Проектирование информационной структуры предметной области

На первом шаге необходимо определить первоначальный объем проекта, т. е. определить информационную структуру, другими словами, выбрать информацию, требуемую для осуществления деятельности.

Для разработки БД учитываются данные, необходимые и достаточные непосредственно для учета техники и оргтехники.

Основные шаги разработки БД:

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		31

1. Определить информационные потребности БД. Проанализировать объекты реального мира, которые необходимо смоделировать в БД. Сформировать из этих объектов сущности и характеристики этих сущностей.

2. Определить поля, которые уникальным образом идентифицируют каждый объект.

3. Установить связи между таблицами.

4. Установить правила ссылочной целостности.

Для создания БД требуется провести анализ и выборку информации, которая находится у работников. Исходными данными являются:

- список техники, находящейся на заводе в данный момент;
- ежемесячные поставки техники;
- отчеты по ремонтам;
- заявки на установку техники пользователям.

В основу разрабатываемой БД необходимо положить серийный номер конкретного вида техники, а также ряд других параметров, которые непосредственно используются на заводе, например, заводской или инвентарный номера техники.

Особенностью предметной области является то, что основными объектами являются комплектующие и периферия.

В системе хранение данных реализовано в виде таблиц четырех групп:

- «Справочники».
- «Составляющие».
- «Периферия».
- «Объекты».

К четвертой группе типа «Объекты» относятся таблицы, такие как «Системный_блок», «Ноутбук» и «Ремонт», каждая из которых, является родительской таблицей для таблиц группы «Составляющие» и «Периферия» и описывается совокупностью классов (сущностей предметной области).

Общей для всех таблицей-справочником является таблица «Производитель», которая связывается со всеми таблицами-справочниками с помощью идентификатора.

					090301.2017.452 ПЗ	Лист
						32
Изм.	Лист	№ докум.	Подпись	Дата		

Информация обо всей технике описывается таблицами-справочниками. К префиксу «Справочник_» добавляется название конкретного комплектующего. Например, в системе определены следующие справочники:

- «Справочник_материнских_плат».
- «Справочник_процессоров».
- «Справочник_принтеров».
- «Справочник_сканеров».
- «Справочник_пользователей» и т. д.

Для каждой из таблиц определен уникальный идентификатор, при помощи которого производится связь таблицы-справочника с соответствующей таблицей «Составляющих» и «Периферия».

Таблицы «Составляющие» содержат сведения обо всех объектах предметной области, являющихся компонентами системного блока, которые имеют свой серийный номер, независимый от серийного номера системного блока, и связываются с основной таблицей «Системный блок» при помощи связи один-ко-многим, при этом идентификатор таблицы «Системный блок» добавляется к таблицам «Составляющие» в виде вторичного ключа. В случае если составляющие могут присутствовать в комплекте системного блока в нескольких экземплярах, например, жесткий диск, то связь между таблицами реализуется при помощи связи многие-ко-многим. Поскольку в реляционной модели баз данных связь многие-ко-многим неразрешима, то для связи таблиц вводится дополнительная таблица, например, в случае таблиц «Системный блок» и «Жесткий диск» дополнительная таблица будет состоять только из двух идентификаторов: идентификатор жесткого диска и идентификатора системного блока (рисунок 3).

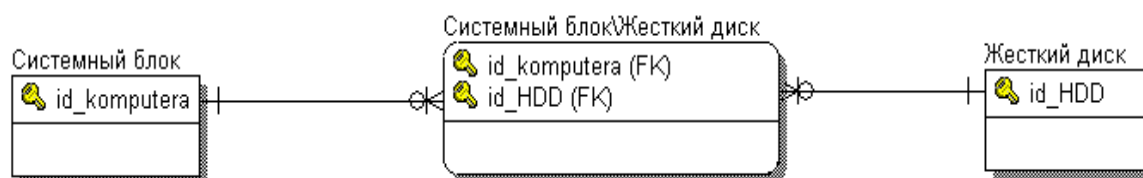


Рисунок 3 – Разрешенная связь многие-ко-многим

Каждый экземпляр техники в конкретный момент времени находится в определенном месте. Каждое место, в котором находится экземпляр техники, долж-

но иметь уникальный идентификатор. Данные о цехе описываются в таблице «Цех».

Таблицы групп «Составляющие», «Периферия» и «Системный блок» и «Ноутбук» связаны с таблицей «Цех» с помощью идентификатора цеха.

Экземпляр техники, находящийся в эксплуатации, характеризуется тем, что находится в конкретном производственном помещении и установлен на соответствующем месте. Поэтому в таблицах «Периферия», «Системный блок» и «Ноутбук» добавлены атомарные поля «Здание» и «Комната».

Кроме того, экземпляр техники, находящийся в эксплуатации, используется определенным пользователем. В свою очередь пользователь обеспечивает контроль над работоспособностью техники.

Информация о пользователе является практически неизменной – данные добавляются или изменяются на основании технических решений при вводе в эксплуатацию новой техники или при модификации, или упразднении имеющихся.

Таблица «Ремонт» описывает регламентированные для каждого экземпляра техники ремонты. Информация этой таблицы отображает:

- дату сдачи в ремонт;
- дату принятия с ремонта;
- сведения о неисправности определенного экземпляра техники;
- место ремонта;
- ответственного за экземпляр техники.

Таблица «Ноутбук» содержит сведения о характеристиках ноутбука. Количество ноутбуков на заводе небольшое, порядка 15 единиц. По этой причине было принято решение не нормализовывать данную таблицу. Основные поля данной таблицы представляют собой символьные поля.

Полная логическая модель БД, содержащая 49 таблиц, приведена в приложении А.

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		34

3.3 Программная реализация АРМ учета и контроля техники и оргтехники

Программная разработка АРМ учета и контроля компьютерной техники и оргтехники включает в себя 3 этапа:

- построение схемы базы данных в ERWin;
- создание базы данных на сервере;
- разработка программного комплекса.

На первом этапе необходимо реализовать в ERWin модель, показанную в приложении А.

Для построения модели данных использовалось CASE-средство ERwin/ERX 4.1.1, позволяющее максимально систематизировать и автоматизировать все этапы разработки программного обеспечения. ERwin – средство разработки структуры базы данных. ERwin сочетает графический интерфейс Windows, инструменты для построения ER-диаграмм, редакторы для создания логического и физического описания модели данных и прозрачную поддержку ведущих реляционных СУБД и настольных баз данных. С помощью ERwin можно создавать или проводить обратное проектирование (реинжиниринг) баз данных [14].

На диаграмме сущность изображается прямоугольником (если зависимая, то с закруглёнными углами). Сущность визуально представляет три вида информации: атрибуты, составляющие первичный ключ, не ключевые атрибуты, тип сущности (независимая/зависимая). Атрибуты, являющиеся первичными ключами, расположены выше горизонтальной черты.

Связь является логическим соотношением между сущностями. При определении связей происходит миграция атрибутов первичного ключа родительской сущности (с признаком (FK)) в соответствующую область атрибутов дочерней сущности. Атрибуты первичного ключа родительской сущности по умолчанию мигрируют со своими именами.

Для связей между таблицами, необходимо установить следующие значения:

- RESTRICT – запрещение удаления;
- CASCADE – каскадное обновление (изменение).

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		35

При переключении между моделями, если физической модели еще не существует, она создается автоматически. Физический уровень представления модели зависит от выбранного сервера. Проектирование на физическом уровне выполнялось в терминах базы данных для СУБД SQL Server 2000. Таблицы базы данных сгенерированы по шаблону на основе соответствующих сущностей и ключей логической модели.

Полученная физическая модель представлена в приложении Б.

На следующем этапе разработки программного комплекса необходимо создать базу данных в серверной части программного комплекса. Для этого, в первую очередь, необходимо создать на имеющемся сервере DBSERVER базу данных с именем s20, а также создать пользователя для программирования и тестирования базы данных и приложения. Кроме того, нужно настроить источник данных с именем s20 для установления соединения ERWin с базой данных s20.

Вначале процесса генерации схемы БД устанавливается сеанс связи ERWin с сервером DBSERVER через псевдоним mla, для чего необходимо ввести имя пользователя и его пароль.

Теперь можно приступить к генерации БД, которую можно осуществить двумя способами. Во-первых, запустить процесс генерации непосредственно в ERWin нажатием кнопки Generate диалога Schema Generation Preview. Во-вторых, сохранить полученный в ERWin файл запросов в SQL текстовом формате и исполнить его, например, с помощью средства Query Analyzer, поставляемого в комплекте с SQL Server.

Заключительным этапом создания АРМ учета компьютерной техники и оргтехники является разработка интерфейса доступа к данным. При разработке внешнего интерфейса пользователя программы была использована объектно-ориентированная библиотека визуальных компонентов (Visual Component Library), являющихся по своей сути «строительными блоками», из которых собирается Delphi-приложение. Она содержит все необходимые для осуществления гибкого и удобного взаимодействия с пользователем компоненты, такие как окна, меню, панели диалогов, кнопки, поля ввода, многочисленные списки, специальные

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		36

компоненты для управления файлами, компоненты для работы с БД и тому подобные средства.

В любом программном продукте, ориентированном на конечного пользователя, немаловажную роль играет простота и понятность интерфейса, ведь непосредственно от этого зависит удобство работы с ним [18].

Для простоты освоения и использования программы все элементы интерфейса выполнены в одном стиле. Так же все функции могут быть вызваны через меню и через определенные комбинации клавиш на клавиатуре.

Основное меню программы содержит четыре пункта. Три из них отображают таблицы-справочники и таблицы хранения данных. Четвертый пункт меню является выходом из программы.

Основное меню программы высвечивается после запуска программы, расположено в верхней части экрана.

Движение по пунктам горизонтального меню происходит с помощью клавиш управления курсором мыши, а также по пунктам выпадающего вертикально меню.

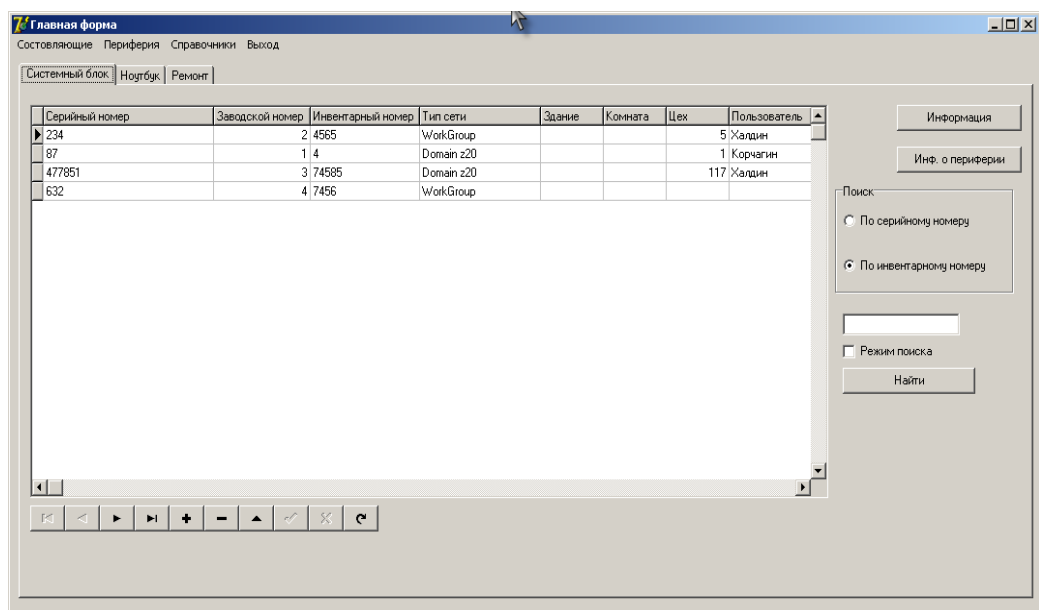


Рисунок 4 – Внешний вид основного меню

Выбор пункта меню осуществляется подсветкой (или с помощью курсора мыши), а запуск – «щелчком» левой кнопки мыши.

Для того чтобы выйти из программы можно выполнить одно из следующих действий:

- «щелкнуть» левой кнопкой мыши по пункту меню «Выход»;
- «щелкнуть» левой кнопкой мыши по кнопке в правом верхнем углу с изображением крестика.

Выбор пункта главного меню «Справочники» производит открытие окна, изображенного на рисунке 5.

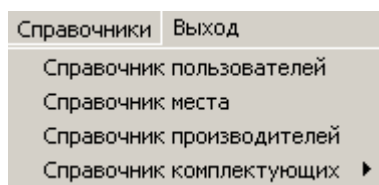


Рисунок 5 – Подменю «Справочники»

В этом подменю выбирается определенный тип объекта:

- справочник пользователей;
- справочник места;
- справочник производителей;
- справочник комплектующих.

По умолчанию установлен режим отображения всех типов объектов. С помощью навигатора, расположенного в левом нижнем углу, возможно добавлять новые записи, редактировать уже существующие записи, удалять при необходимости, и передвигаться по ним.



Рисунок 6 – Навигатор

Активизировать выше перечисленные операции можно непосредственным нажатием левой кнопкой мыши соответствующей кнопки.

Справочники являются наиважнейшей составной частью БД. Также они обеспечивают стандартизацию и унификацию представления в БД основных учетных характеристик. Для просмотра, добавления, удаления и редактирования записей в справочниках используется пункт меню «Справочники».

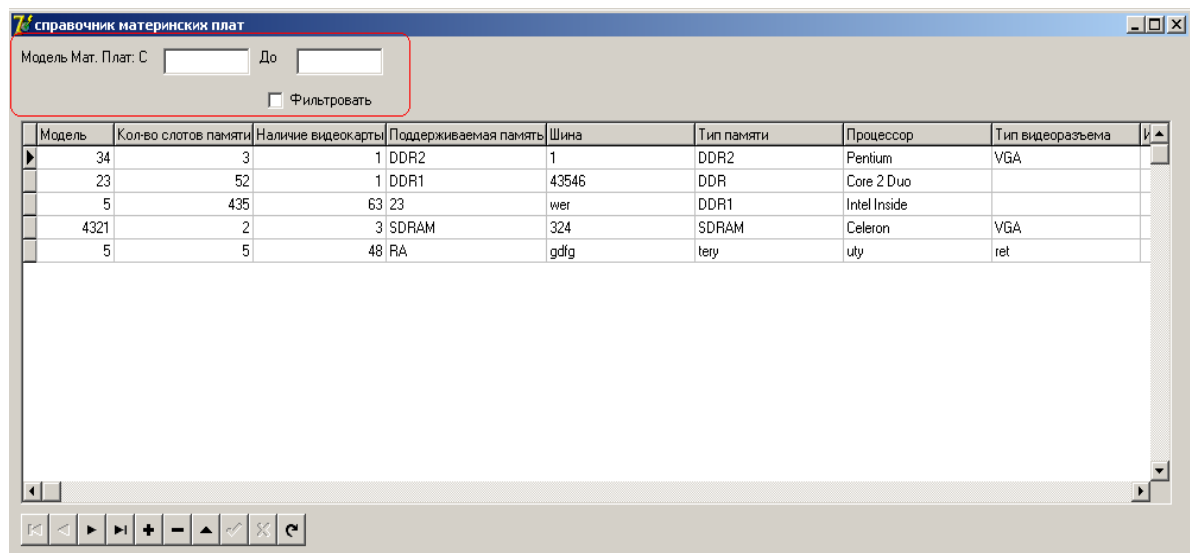


Рисунок 7 – Форма «Справочник материнских плат»

Справочники заполняются и пополняются администратором базы данных вручную.

Все изменения, вносимые в справочник, автоматически вносятся в соответствующие характеристики представленных в базе объектов.

Выбор пункта главного меню «Составляющие» производит открытие подменю, изображенного на рисунке 8.

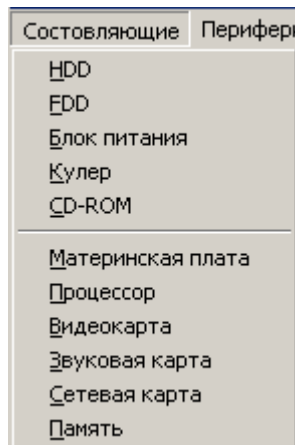


Рисунок 8 – Подменю «Составляющие»

В этом меню представлены комплектующие компьютера, эксплуатируемые на предприятии.

При выборе одного из этих пунктов открывается соответствующая форма для просмотра данных.

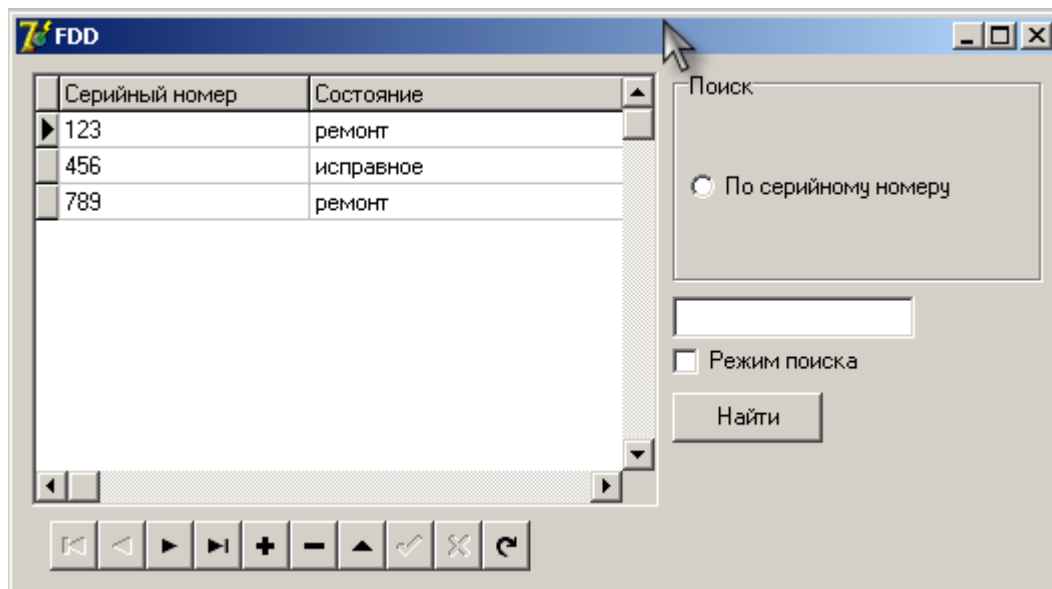


Рисунок 9 – Форма FDD

Для поиска записей в таблицах данных используется поиск, реализованный с помощью визуальных компонентов. Поиск реализован на каждой форме в правом верхнем углу от основной таблицы (рисунок 10).

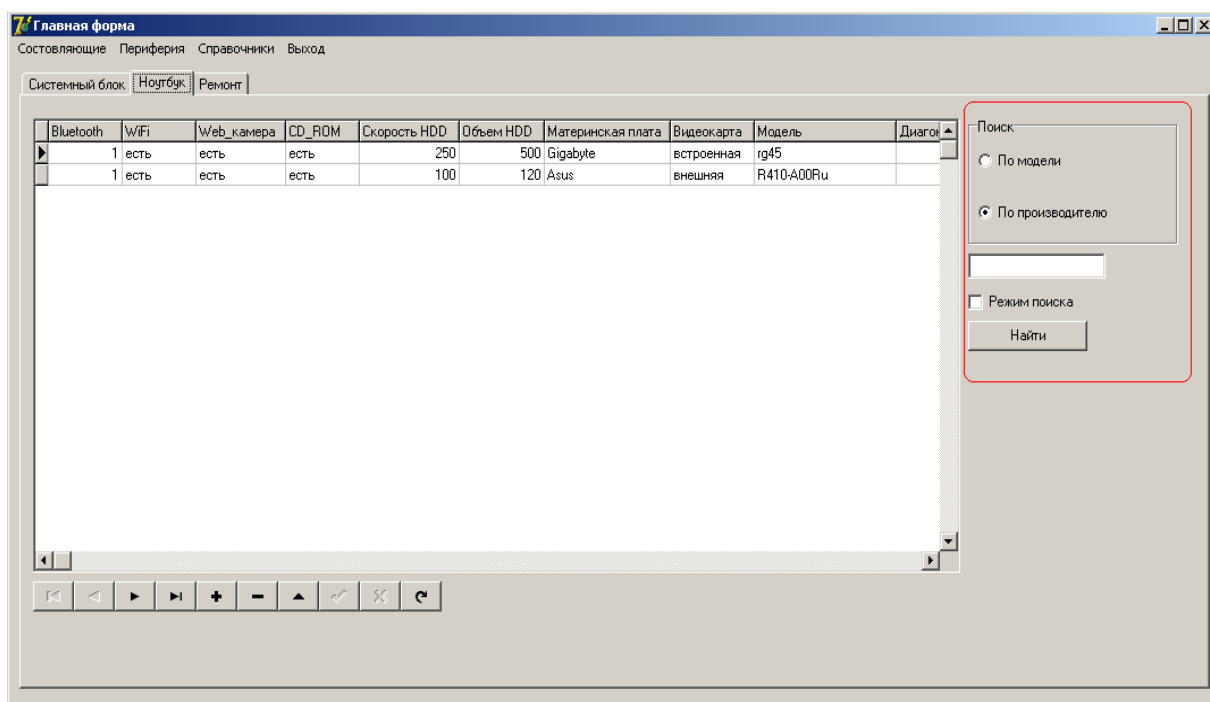


Рисунок 10 – Поиск

Поиск возможен по различным критериям, например, поиск по серийному или инвентарному номерам. При выборе одного из критерия поиска выполняется поиск по первым символам, вводимым в строку поиска. Для активации режима поиска необходимо поставить галочку в поле «Режим поиска».

Так же в таблицах-справочниках для большей простоты поиска, были реализованы фильтры (рисунок 11).



Рисунок 11 – Фильтр

Фильтр реализован при помощи визуальных компонентов и расположен сверху над главной таблицей формы. Фильтрация записей осуществляется вводом диапазона в нужные поля и в таблице отображается строго только указанный диапазон значений.

Реализовано несколько вариантов фильтрации записей:

1. вводится диапазон значений, записи выводятся в указанном диапазоне от меньшего к большему, в случае если первое значение меньше второго и от большего к меньшему в противном случае;
2. вводится только одно значение диапазона от начального, информация выводится в таблицу начиная с того значения, которое ввели в строку;
3. вводится последнее значение диапазона поиска, записи выводятся все до вводимого значения;
4. если записи, введенные в оба диапазона равны, то выводятся записи равные значению диапазона, записей может быть либо одна, либо несколько.

На главной форме приложения отображаются сведения о:

- системных блоках;
- ноутбуках;
- ремонтах техники.

Для просмотра информации о системном блоке необходимо выбрать вкладку «Системный блок» и нажать кнопку «Информация».

Данная форма (рисунок 12) выводится после нажатия кнопки «Информация» расположенной справа от главной таблицы на вкладке «Системный блок». На форме «Информация» сводятся данные, взятые из нескольких таблиц реализованных при помощи SQL запроса в компоненте Query. Информация о компонентах, составляющих системный блок, выводится в виде

таблиц, поскольку в составе системного блока могут находиться несколько составляющих одного типа. Например, оперативная память или жесткий диск.

The screenshot shows a software interface for configuring a computer system. It lists various hardware components with their respective specifications and a 'Выбрать' (Select) button for each. The components and their details are as follows:

Компонент	Сериальный номер	Заводской номер	Модель	Кол-во слотов памяти	Наличие видеокарты	Поддерживаемая память	Шина	Тип памяти
Материнская плата	234	2	34	3	1	DDR2	1	DDR2
Процессор	45645	423	4	в эксплуатации				
Память								
Видеокарта	Samsung	456156	23	653	в эксплуатации			
Звуковая карта	Samsung	543	54646					
Сетевая карта	Samsung	345	12	DDE			123	в
Блок питания	Samsung	12	12				180	sd
Винчестер	Samsung	55555	234	34	45	erte		
CD-ROM	Samsung	54	2	на складе				
FDD	Samsung	123	12	i				
Кулер	Samsung	54234	n4	12	используется			

Рисунок 12 – Информация о системном блоке

Аналогичная форма «Информация о периферии» выводит данные о подключенной периферии к данному компьютеру. Данная форма так же реализована при помощи SQL запроса компонента Query.

Периферия							
Монитор	Наименование производителя	Серийный номер	Заводской номер	Инв.			Выбрать
	Samsung	4534	543	2534			
Клавиатура	Наименование производителя	Серийный номер	Заводской номер	Мод.			Выбрать
	Samsung	213	12				
Мышь	Наименование производителя	Серийный номер	Заводской номер	Мод.			Выбрать
	Samsung	1234	13445				
Колонки	Наименование производителя	Серийный номер	Заводской номер	Инв.			Выбрать
	Samsung	45	23	423			
Принтер	Наименование производителя	Серийный номер	Заводской номер	Инвентарный номер	Модель	Тип п.	Выбрать
	HP	12	12	12	23423	laser	
Сканер	Наименование производителя	Серийный номер	Заводской номер	Инвентарный номер	Модель		Выбрать
	Samsung	45	12	43	п4		
Песперебойник питания	Наименование производителя	Серийный номер	Модель	Количество розеток	Состояние		Выбрать
	Samsung	21	23	4	123		
Контроллер	Наименование производителя	Серийный номер	Заводской номер	Инвентарный номер	Модель	Дс.	Выбрать
	Pittal	42123	633	983476	апрк5	иог	
Кард Ридер	Наименование производителя	Серийный номер	Модель	Состояние			Выбрать
	Samsung	1234	54	d1hdh			

Рисунок 13 – Информация о периферии

Выбор компьютера, о котором пользователь хочет получить информацию, производится выделением курсором мышки записи в таблице «Системный блок».

Данные на этих формах «Информация» и «Информация о периферии» нельзя изменить, дополнить или удалить, т. к. данные берутся из соответствующих таблиц.

Что бы добавить к системному блоку какой-либо вид техники, будь то жесткий диск или сканер, на вспомогательных таблицах вывода информации напротив каждого вида техники есть кнопка «Выбрать».

Кулер	Наименование производителя	Серийный номер	Модель	Размер	Состояние	Выбрать
	Samsung	54234	п4	12	используется	

Рисунок 14 – Кнопка «Выбрать»

После ее нажатия открывается соответствующая форма данного вида техники, производится выделение нужной записи, после чего необходимо нажать

кнопку «Добавить», расположенной на этой же форме. Кнопка «Добавить» видна только в режиме выбора вида техники.

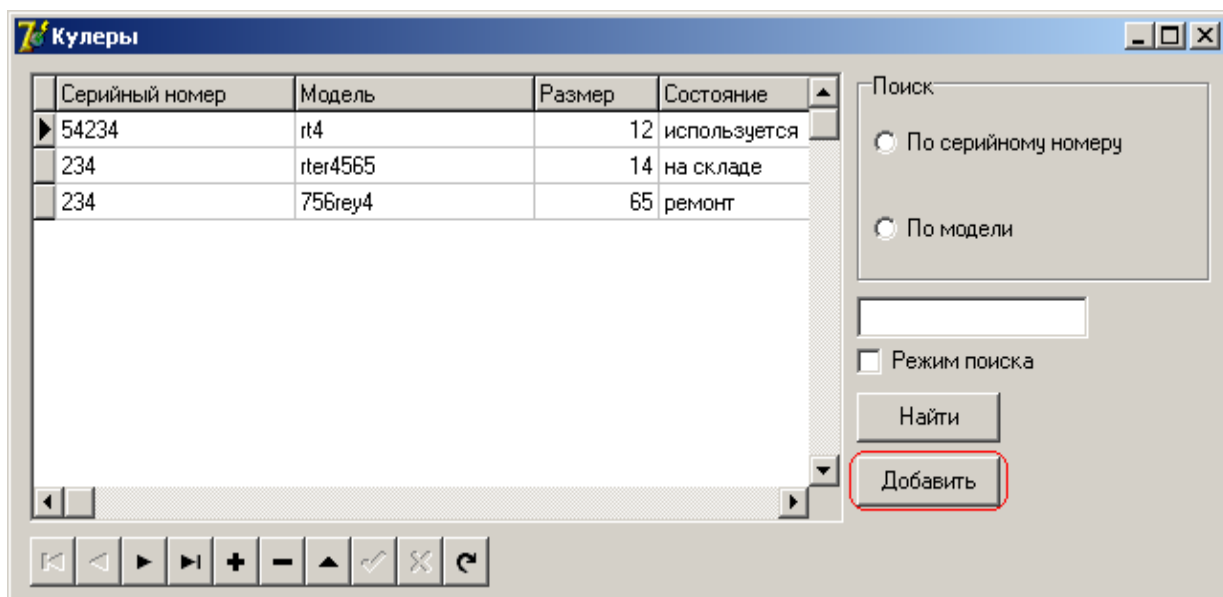


Рисунок 15 – Кнопка «Добавить»

После чего, идентификатор системного блока будет добавлен в соответствующую таблицу. Данная процедура добавления реализована при помощи SQL запроса.

В программе реализована возможность распечатки технического паспорта ПЭВМ. Это реализовано про помощи компонента RaveReport. На вкладке «Системный блок» имеется кнопка «Распечатать тех паспорт».

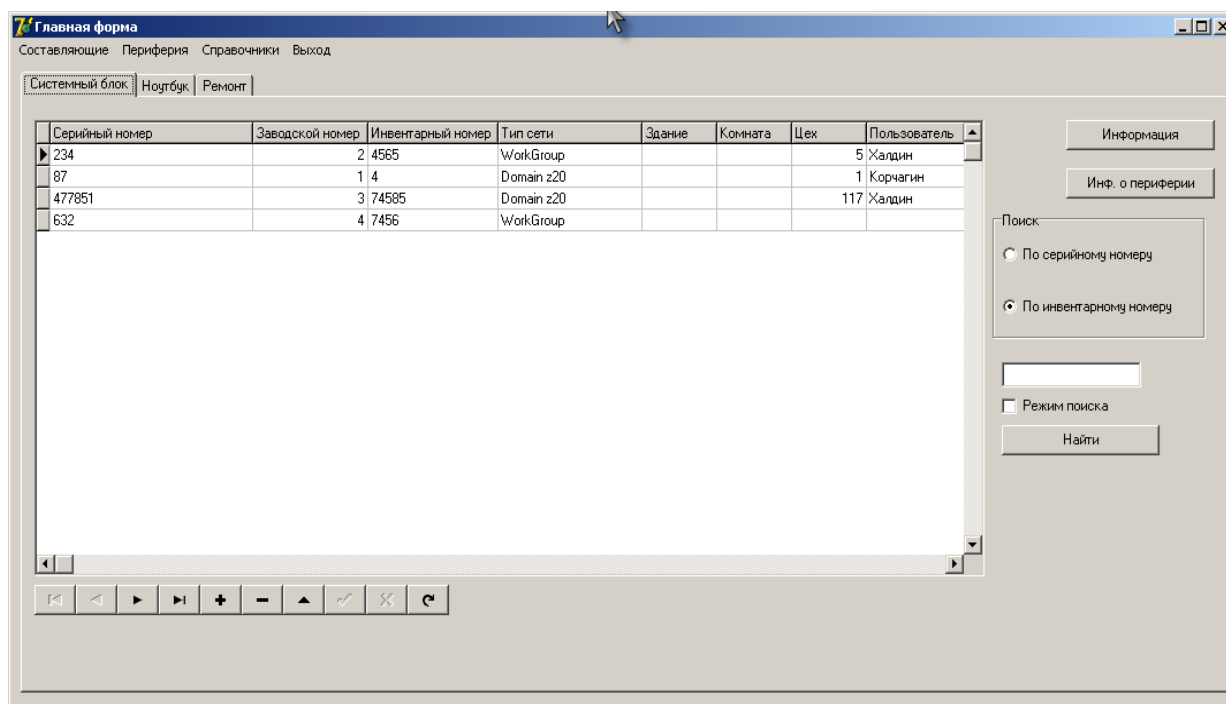


Рисунок 16 – Кнопка печати

При ее нажатии открывается форма отчета для просмотра технического паспорта на экране.

Паспорт на персональную ЭВМ

<u>Серийный номер</u>	87
<u>Заводской номер</u>	1
<u>Инвентарный номер</u>	4

Сведения о пользователе		
Пользователь	Место	Телефон
Корчагин Евгений Юрьевич	1/1/14	

Подпись _____

Рисунок 17 – Пример отчета

Операция распечатки документа выполняется стандартными средствами компонента RaveReport. Изменить форму отчета может только администратор базы данных путем перекомпиляции проекта в Delphi.

На вкладке «Ноутбук» отображается таблица, хранящая в себе все сведения о ноутбуках. Данные описывают все характеристики определенной модели.

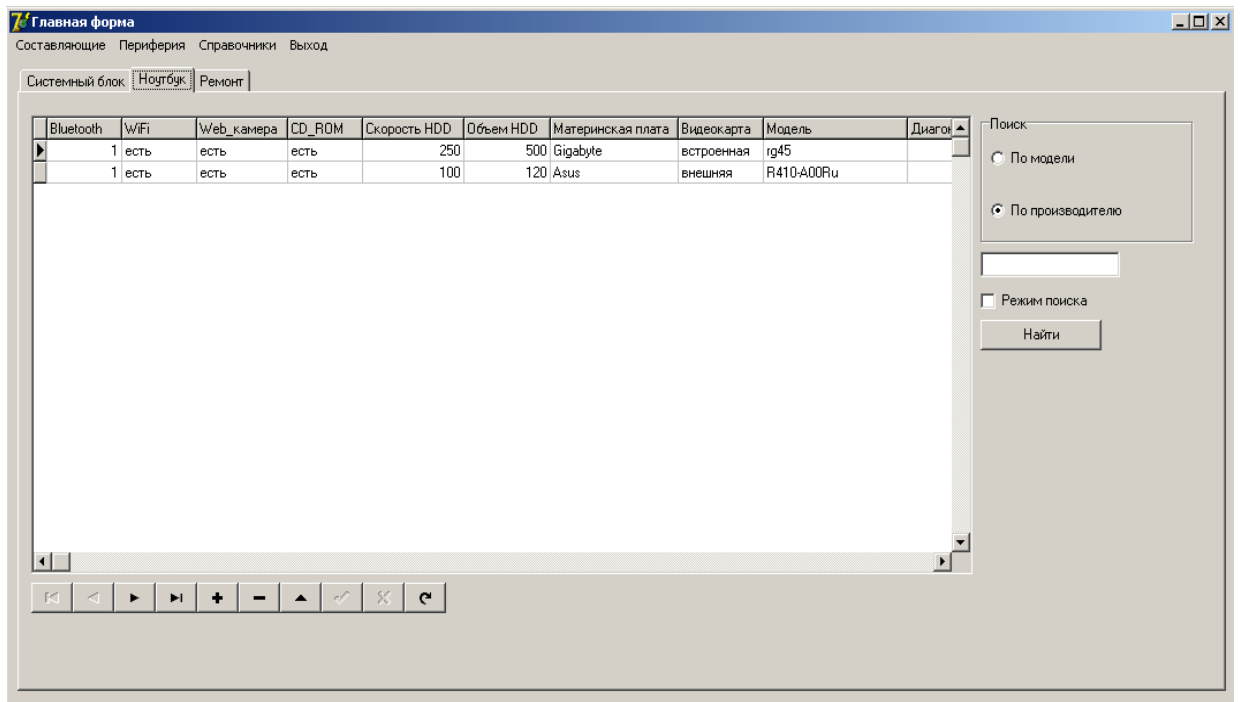


Рисунок 18 – Вкладка «Ноутбук»

Вкладка «Ремонт» отображает сведения о видах техники, отправляемых и получаемых с ремонта. Также описывается какой именно вид техники неисправен и вид неполадки.

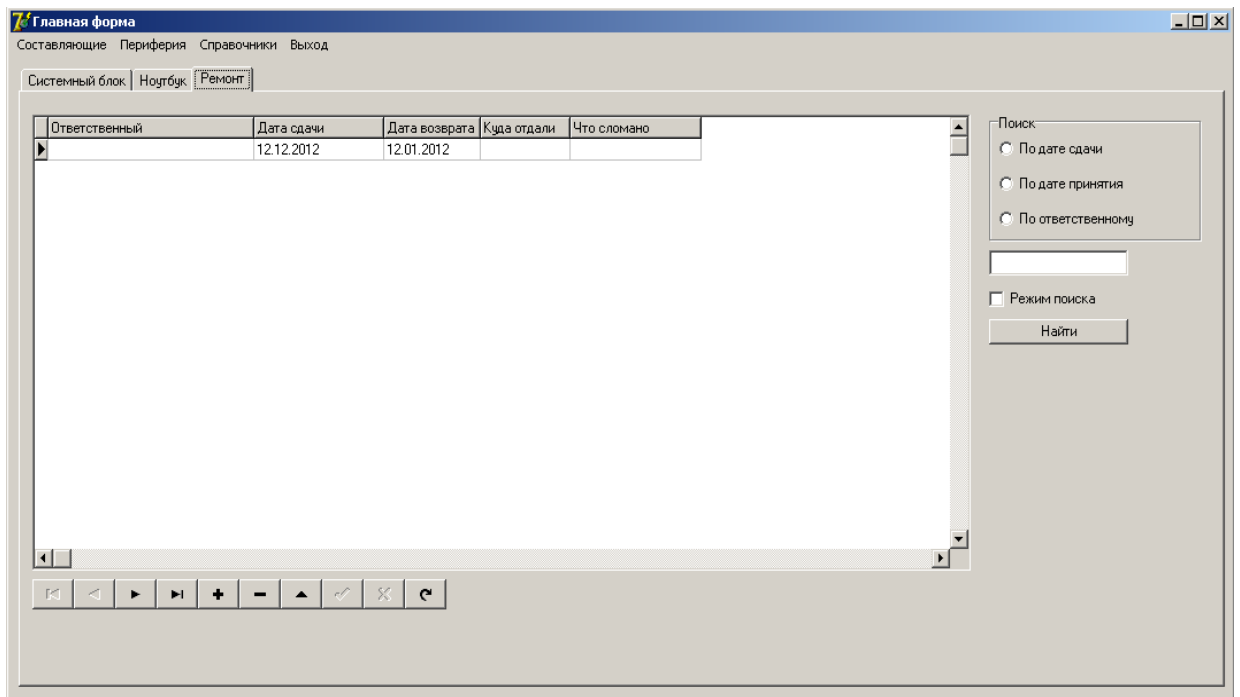


Рисунок 19 – Вкладка «Ремонт»

Итак, пользователь при работе с программой имеет возможность:

- редактировать, производить поиск, просматривать, сортировать данные обо всех экземплярах техники;
- оперативно получать данные обо всей технике, находящейся в данный момент на ремонте;
- просматривать содержание конкретного устройства в определенном компьютере;
- получать данные о местонахождении техники;
- просматривать данные о пользователе, закрепленном за определенным видом техники;
- получать справочную информацию о технике.

Программа обеспечивает одновременный доступ многих пользователей к одним и тем же данным, с использованием одних и тех же правил и ограничений.

Программа имеет удобный интуитивный интерфейс, проста в освоении, не требовательна к аппаратным ресурсам.

В процессе создания АРМ проводились консультации с будущими пользователями. Были учтены все пожелания.

3.4 Расчет технико-экономических показателей разработки программного комплекса

Для того чтобы произвести необходимую обществу продукцию, подготовить ее к реализации, отгрузить заказчику, предприятие расходует средства на оплату труда рабочих, инженерно-технических работников и служащих, приобретение материалов, топлива и энергии, содержание и эксплуатацию оборудования, зданий и сооружений. Сумма денежных затрат предприятия на производство и реализацию продукции называется ее себестоимостью [22].

Для достижения на всех предприятиях единства в планировании и учете однотипных затрат, обеспечения строгой регламентации расходов и контроля за ними принято группировать затраты следующим образом:

- по экономическим элементам;
- по калькуляционным статьям.

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		47

Элементы затрат представляют первичные, однородные по своему экономическому содержанию группы расходов, которые не могут быть разложены на какие-либо более простые слагаемые, различающиеся по содержанию. По каждому элементу объединяются экономически однотипные расходы независимо от места затрат (цех, заводоуправление и т.п.), объекта затрат (изготовление продукции, отгрузка потребителю продукции и др.). Затраты на производство продукции всей промышленности, отдельных отраслей и предприятий распределяются по следующим элементам:

C1 – сырье и основные материалы, в том числе покупные комплектующие изделия, за вычетом возвратных отходов;

C2 – вспомогательные материалы;

C3 – топливо со стороны (покупное): технологическое топливо и для бытовых нужд топливо;

C4 – энергия со стороны (технологическая и бытовая);

C5 – зарплата основная и дополнительная из фонда заработной платы;

C6 – начисления на зарплату (по нормативам);

C7 – амортизация основных фондов;

C8 – прочие денежные расходы.

Полная себестоимость определяется по формуле:

$$C = \sum_i C_i \quad (1)$$

Первые семь элементов полностью однородны по своему экономическому содержанию. Последняя группа, условно называемая элементом затрат «прочие денежные расходы», составляет исключение. В эту группу включаются различные по экономическому содержанию затраты; общим признаком для них является невозможность включения в какой-либо из элементов затрат. В остальных элементах какое-либо смешение разнородных затрат не допускается.

Группировка затрат по экономическим элементам необходима для составления сметы затрат на производство, определения общей суммы затрат, связанных с производством продукции. Это имеет большое значение не только для предприятий, но и для многих народнохозяйственных расчетов, например, для

					090301.2017.452 ПЗ	Лист
						48
Изм.	Лист	№ докум.	Подпись	Дата		

определения чистой продукции промышленности и национального дохода страны в целом.

Каждое производственное объединение (предприятие) должно окупать свои затраты на изготовление продукции и получать прибыль – работать рентабельно.

Прибыль образуется вследствие того, что объединение и предприятие реализуют свою продукцию не по ее себестоимости, а по оптовым ценам. Цена представляет денежное выражение стоимости продукта. В цене находят отражение общественно необходимые затраты на производство данного товара и чистый доход общества (прибыль и налог с оборота).

Между ценой, себестоимостью и прибылью существует тесная взаимосвязь. Чем ниже себестоимость, тем выше прибыль, тем эффективнее работает предприятие, а, следовательно, будет иметь выше уровень рентабельности.

Ниже приведен расчет себестоимости программного продукта.

Расчет затрат на материалы.

Для выполнения выпускной квалификационной работы необходимы средства, предоставляемые предприятием (ОС, среда разработки приложений, текстовый редактор, СУБД, CASE-средство), и покупаемые непосредственно для решения данной задачи (бумага формата А4 и картридж для принтера HP LaserJet 4100n). Результаты расчетов затрат на покупные изделия сведены в таблицу 1.

					090301.2017.452 ПЗ	Лист
						49
Изм.	Лист	№ докум.	Подпись	Дата		

Таблица 1 – Затраты на покупные изделия

Материалы	Единицы измерения	Количество	Цена за единицу, руб.	Сумма, руб.
Программные средства:				
Windows XP Professional	Прогр. продукт	1	4 056	4 056
Borland Delphi Enterprise 7.0	Прогр. продукт	1	55 239	55 239
Microsoft Office 2007	Прогр. продукт	1	3 744	3 744
Microsoft SQL Server 2000	Прогр. продукт	1	25 700	25 700
Platinum Erwin/ERX 4.1.4	Прогр. продукт	1	67 884	67 884
Расходные материалы:				
Flash-накопитель 16 Gb	Шт.	1	1 200	1 200
Бумага формата А4	Пачка	1	120	120

Окончание таблицы 1

Материалы	Единицы измерения	Количество	Цена за единицу, руб.	Сумма, руб.
Матрица CD-R	Шт.	2	15	30
Картридж	Шт.	1	1 860	1 860
Транспортно-заготовительные расходы:	%	0,1	-	1 646
Итого:				161 479

* – указанные программные продукты использованы ориентировочно для 10 задач. Поэтому для расчета учитываем 10 % от стоимости данных продуктов.

Общая стоимость затрат на вспомогательные материалы составляет 21 003 руб.

					090301.2017.452 ПЗ	Лист
						50
Изм.	Лист	№ докум.	Подпись	Дата		

Расчет затрат на потребляемую электроэнергию.

Стоимость потребляемой электроэнергии определяется по формуле:

$$Z_{эл} = W \cdot T \cdot S_{эл} \text{ руб.}, \quad (2)$$

где W – Установленная электрическая мощность, кВт;

T – Число часов работы оборудования в течении времени разработки программы;

$S_{эл}$ – тариф за электроэнергию, руб./кВт * час.

Расчет стоимости потребляемой электроэнергии приведен в таблице 2.

Таблица 2 – Стоимость потребляемой энергии

Наименование оборудования	Кол-во, Шт.	Мощность, кВт	Время работы, час.	Тариф на электроэнергию, руб./кВт*час	Стоимость энергии, руб.
Компьютер PENTIUM III	1	0,3	560	2,24	376,32
Принтер HP LaserJet 4100n	1	0,22	5	2,24	2,46
Итого:					378,78

Стоимость энергозатрат за период создания программного продукта составляет 378 руб. 78 коп.

Расчет заработной платы.

Заработная плата состоит из следующих статей:

- оклад;
- премия;
- районный коэффициент.

Расчет основной заработной платы сведен в таблицу 3.

Таблица 3 – Расчет основной заработной платы

Виды работ	Категория работников	Продолжительность работы, мес.	Должностной оклад	Премия 40%, руб.	Районный коэффициент 30%, рублей	Итого зарплата, руб.	
						за 1 месяц	за период работ
Разработка программного продукта	Инженер	3	14 000	5 600	5 880	25 480	76 440

Заработная плата за три месяца составляет 76 440 руб.

Единый социальный налог.

Единый социальный налог составляет 26,5 % от заработной платы и равняется 20 256,6 рубль.

Расчет амортизационных отчислений.

Амортизационные отчисления на оборудование за время работы на нем определяются следующим образом:

$$A = \frac{C_6 N_a}{100} \cdot \frac{T}{12} \text{ руб.}, \quad (3)$$

где C_6 – балансовая стоимость оборудования, руб.;

N_a – годовая норма амортизации, %;

T – время работы на данном оборудовании, мес.

Расчет расходов на амортизацию приведен в таблице 4.

Таблица 4 – Расходы на амортизацию

Наименование оборудования	Балансовая стоимость, руб.	Годовая норма амортизации, %	Амортизационные отчисления, руб.	
			за год	за период работ месяца
Компьютер PENTIUM III	28 375	20	5 675	1 418,75
Принтер HP LaserJet 4100n	11 200	20	2 240	560
Итого:				1 978,75

Расчет накладных расходов.

Накладные расходы включают в себя общепроизводственные и общехозяйственные расходы.

Общепроизводственные расходы составляют 92 % от затрат на материалы, энергию (энергозатраты), заработной платы, единого социального налога и амортизационных отчислений (от суммы перечисленных показателей). Общехозяйственные расходы составляют 20 % от суммы всех затрат.

Сводная калькуляция проведенных расчетов приведена в таблице 5.

Таблица 5 – Сводная калькуляция

№ п/п.	Статьи расходов	Расчет	Стоимость, руб.
1.	Материалы	табл. 1	21 003
2.	Энергозатраты.	табл. 2	378,78
3.	Заработная плата	табл. 3	76 440
4.	Единый социальный налог	26,5 % от п.3	20 256,6
5.	Амортизационные отчисления	табл. 4	1 978,75
6.	Общепроизводственные расходы	92 % от \sum (п1-5)	104 587
7.	Цеховая себестоимость	\sum (п.1 - 6)	218 268,53
8.	Общехозяйственные расходы	20 % от п.7	43 653,7
9.	Полная себестоимость	п.7 + п.8	268297,83

Таким образом, полная себестоимость программного продукта составляет 268 297 руб. 83 коп., что, прежде всего, обусловлено дороговизной используемого при проектировании ПО. Однако в целях сокращения времени для разработки

информационной модели, построения логических диаграмм и диаграмм данных, генерации скриптов БД средствами Erwin, сокращение трудозатрат, требуемые для учета техники и оргтехники, за счет упрощения операции пополнения и представления информации по ремонтам, добавлениям, использованию и отказам, постоянного контроля за корректностью вводимой информации, упрощает работу, исключает ошибки, часто встречающиеся при обычной организации работы, ориентирован на конечного пользователя, простота и понятность интерфейса, создание уникального программного комплекса для отдела главного прибориста, в котором едино и неразрывно осуществляется проводимый в настоящее время контроль и учёт техники, а также необходимость создания подобного рода программного комплекса, оправдывает данные затраты.

					090301.2017.452 ПЗ	Лист
						54
Изм.	Лист	№ докум.	Подпись	Дата		

ЗАКЛЮЧЕНИЕ

Цель данной выпускной квалификационной работы состояла в разработке автоматизированного рабочего места учета и контроля компьютерной техники и оргтехники на заводе. Этот программный комплекс представляет собой продукт, в котором едино и неразрывно осуществляется проводимый в настоящее время учет техники.

Основной задачей является ведение электронной БД обо всех типах техники, используемых на предприятии.

Проект работает в режиме реального времени – это означает, что соответствующие лица обеспечивают своевременное оформление данных (обычно в пределах рабочего дня) обо всех операциях внесения, удаления или изменения.

В результате проделанной работы создан программный комплекс, который автоматизирует деятельность предприятия. Его использование упрощает процесс учета данных о технике, а также ведение журналов по ремонтам. Разработанный проект тестируется и в дальнейшем будет внедрен в использование на предприятии.

					090301.2017.452 ПЗ	Лист
						55
Изм.	Лист	№ докум.	Подпись	Дата		

СПИСОК СОКРАЩЕНИЙ И ТЕРМИНОЛОГИЯ

БД – База Данных.

ГИП – Графический Интерфейс Пользователя.

ОГП – Отдел Главного Прибориста.

ОС – Операционная Система.

ПО – Программное Обеспечение.

ППР – Планово-Предупредительные Работы.

СУБД – Система Управления Базами Данных.

СУРБД – Система Управления Распределенными Базами Данных.

ТС – Технологическая Система.

API – Application Programming Interface.

SQL – Structured Query Language (структурированный язык запросов).

АРМ – Автоматизированное Рабочее Место.

АСУТП – Автоматизированная Система Управления Технологическим Процессом.

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		56

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

- 1 Артемов, Д. Microsoft SQL Server 7.0: установка, управление, оптимизация / Д. Артемов, Г. Погульский. – М.: Издательский отдел «Русская Редакция» ТОО «Channel Trading Ltd.», 1998.
- 2 Баас, Р. Delphi 4: полное руководство: пер. с нем. / Р. Баас, М.: Фервай, Х. Гюнтер. – Киев: Издательская группа ВНУ, 1998.
- 3 Васкевич, Д. Стратегии клиент/сервер. Руководство по выживанию для специалистов по реорганизации бизнеса / Д. Васкевич. – К.: Диалектика, 1996.
- 4 Возневич, Э. Delphi. Освой самостоятельно: Пер. с англ./ Э. Возневич. – М.: Восточная Книжная Компания, 1996.
- 5 Глушаков, С.В. Программирование в среде Windows: Учебный курс/ С.В. Глушаков, С.В. Мельников. – Харьков: Фолио; М: ООО Издательство АСТ», 2000.
- 6 Гофман, В.Э. Delphi 6 / В.Э. Гофман, А.Д. Хомоненко. – СПб.: БХВ-Петербург, 2001.
- 7 Дарахвелидзе, П.Г. Программирование в Delphi 5/ П.Г. Дарахвелидзе, Е.П. Марков, О.А. Котенок. – СПб.: БХВ - Санкт-Петербург, 2000.
- 8 Дантемманн, Д. Д. Программирование в среде Delphi / Д. Дантемманн, Д. Мишел. – К: Издательство «ДиаСофт», 1997.
- 9 Диго, С.М. Проектирование баз данных. Москва: Финансы и статистика / С.М. Диго. – М.: Издательство «Финансы и статистика», 1988.
- 10 Кларк, Д.Дж. IV. Руководство Novell для специалистов CNE в 2^х томах / Д.Дж. Кларк IV. – К: Издательство «ЛЮРИ», 1996.
- 11 Маклаков, С. ВРWin и ERWin: CASE-средства разработки информационных систем / С. Маклаков. – М.: «Диалог-МИФИ», 2000.
- 12 Мамаев, Е.В. Microsoft SQL Server 2000 / Е.В. Мамаев. – СПб.: БХВ-Петербург, 2004.
- 13 Шпеник, М. Руководство администратора баз данных Microsoft SQL Server 7.0.: Пер. с англ / М. Шпеник, О. Следж. – М.: Издательский дом «Вильямс», 1999.

					090301.2017.452 ПЗ	Лист
Изм.	Лист	№ докум.	Подпись	Дата		57

- 14 Грабер, М. Справочное руководство по SQL. Пер. с англ / М. Грабер. – М: Издательство «ЛОРИ», 1997.
- 15 Минаси, М. Графический интерфейс пользователя: секреты проектирования / М. Минаси. – Москва: Мир, 1996.
- 16 Рихтер, Дж. Windows для профессионалов: Программирование для Windows 95 и Windows NT на базе Win32/ Пер. с англ / Дж. Рихтер. – М.: Издательский отдел «Русская редакция» ТОО «Channel Trading Ltd», 1997.
- 17 Сван, Т. Delphi 4. Библия разработчика.: Пер. с англ / Т. Сван. – СПб.: Диалектика, 1998.
- 18 Сибаров, Ю.Г. Охрана труда в вычислительных центрах / Ю.Г. Сибаров, В.Н. Сколотнев. – М.: Машиностроение, 1990.
- 19 Беклешова, В.К. Техничко-экономическое обоснование дипломных работ / В.К. Беклешова. – М.: Высшая школа, 1998.
- 20 Фаронов, В.В. Delphi 4 Руководство по базам данных / В.В. Фараонов, П.В. Шумаков. – М.: «Нолидж», 1999.
- 21 Шумилин, В.К. Краткий курс безопасности / В.К. Шумилин. – М.: Издательство «СОУЭЛО», 2002.
- 22 DVB-RCS – Product Description, EMC TECHNOLOGIES. – Канада, 2004.
- 23 Genov, A. The conception of constructing global spread-spectrum CDMA mobile telecommunication «Global-SS» system. Forum of the ITA Proceedings / A. Genov, N. Ivanchuk. – Moscow, 1997.
- 24 Sky Edge – Product Description, GILAT. – Израиль, 2004.
- 25 Автоматизированное рабочее место в системе управления предприятием: сборник научных трудов. – Ленинград, 1989. – 236 с.
- 26 Агуров, П.В. Интерфейс USB Практика использования и программирования / П.В. Агуров. – М.: ВHV, 2007. – 576 с.
- 27 Александров, Е.А. Основы теории эвристических решений. Подход к изучению естественного и построению искусственного интеллекта / Е.А. Александров. – М.: Советское радио, 1975. – 254 с.

					090301.2017.452 ПЗ	Лист
						58
Изм.	Лист	№ докум.	Подпись	Дата		

- 28 Александров, А.Г. Оптимальные и адаптивные системы / А.Г. Александров. – М.: Высш.шк., 1989. – 263 с.
- 29 Бесекерский, В.А. Теория систем автоматического регулирования / В.А. Бесекерский, Е.П. Попов. – М.:Физматгиз, 1975. – 768 с.
- 30 Сборник задач по теории автоматического регулирования и управления / под ред. В.А. Бесекерского. – М.: Наука, 1978. – 512 с.
- 31 Амалия, М. Архитектура ЭВМ и искусственный интеллект / М. Амалия, Ю. Танака. – М.: Мир, 1993. – 397 с.
- 32 Искусственный интеллект / под ред. Э.В. Попова. – М.: Радио и связь, 1990. – 573 с.
- 33 Банди, Б. Основы линейного программирования / Б. Банди. – М.: Радио и связь, 1989.
- 34 Методологические основы и математические методы / под ред. Дж. Моудера, С. Элмаграби. – М.: Мир, 1981. – Т.1. – 712 с.
- 35 Бежанов, А. Диагностика инжекторных двигателей / А. Бежанов // Системы безопасности. – № 1 (92), 2007.
- 36 Берзин, Е.А. Оптимальное распределение ресурсов и элементы синтеза систем / Е.А. Березин. – М.: Сов. радио, 1974. – 303 с.
- 37 Боресков, А.В. Основы работы с технологией CUDA / А.В. Боресков, А.А. Харламов. – М.: ДМК Пресс, 2010. – 232 с.
- 38 Генов, А. Мультисервисные БЦП – технологический прорыв в повышении эффективности ССС / А. Генов // Науч.-технич. конф.: К 75-летию академика В.А. Мельникова. – М., 2003. – С. 128–236.
- 39 Генов, А. Бортовые цифровые платформы / А. Генов // Broadcasting, 2002. – № 3.
- 40 Генов, А. Исследование вопросов выбора канальной емкости пучков СПСС двойного назначения / А. Генов // Науч.-технич. конф.: Оптические, сотовые и спутниковые сети и системы связи. – Псков, 1996.
- 41 Генов, А. Методика моделирования систем спутниковой связи / А. Генов, В. Голованов // Вопросы радиоэлектроники. – 1978. – № 9.

					090301.2017.452 ПЗ	Лист
						59
Изм.	Лист	№ докум.	Подпись	Дата		

42 Генов, А.О. Влияние характеристик входящего потока старшего приоритета на показатели эффективности функционирования малоканальных СМО с ожиданием / А. Генов, В. Ермилов. – М.: Наука, 2007. – 238 с.

43 ГОСТ 27514-87. Методы расчёта в электроустановках переменного тока напряжением 1 кВ. – М.: Изд-во стандартов, 1988. – 6 с.

44 Громкович, Ю. Теретическая информатика. Введение в теорию автоматов, теорию вычислимости, теорию сложности / Ю. Громкович. – М.: BNV, 2010. – 336 с.

45 Джейсуол, Н. Очереди с приоритетами / Н. Джейсуол. – М.: Мир, 1973.

46 Егоршин, А.Ю. Новая система управления реального времени мобильного объекта / А.Ю. Егоршин. – М.: Роспатент, 2006.

47 Егоршин, А.Ю. Новости на основе нечетких когнитивных карт / А.Ю. Егоршин // Интеллектуальные системы обработки информации и управления: сб. ст. Рег. зимн. шк.–сем. аспирантов и молодых ученых. – Уфа: Технология, 2006. – Т. 2. – С. 89–97.

48 Ефимова, М.Р. Общая теория статистики: учебник / М.Р. Ефимова. – М.: ИНФРА-М, 1998. – 368 с.

49 Заботина, Н.Н. Проектирование информационных систем: учебное пособие / Н.Н. Заботина. – М.: ИНФРА-М, 2011. – 331 с.

					090301.2017.452 ПЗ	Лист
						60
Изм.	Лист	№ докум.	Подпись	Дата		

ПРИЛОЖЕНИЕ А

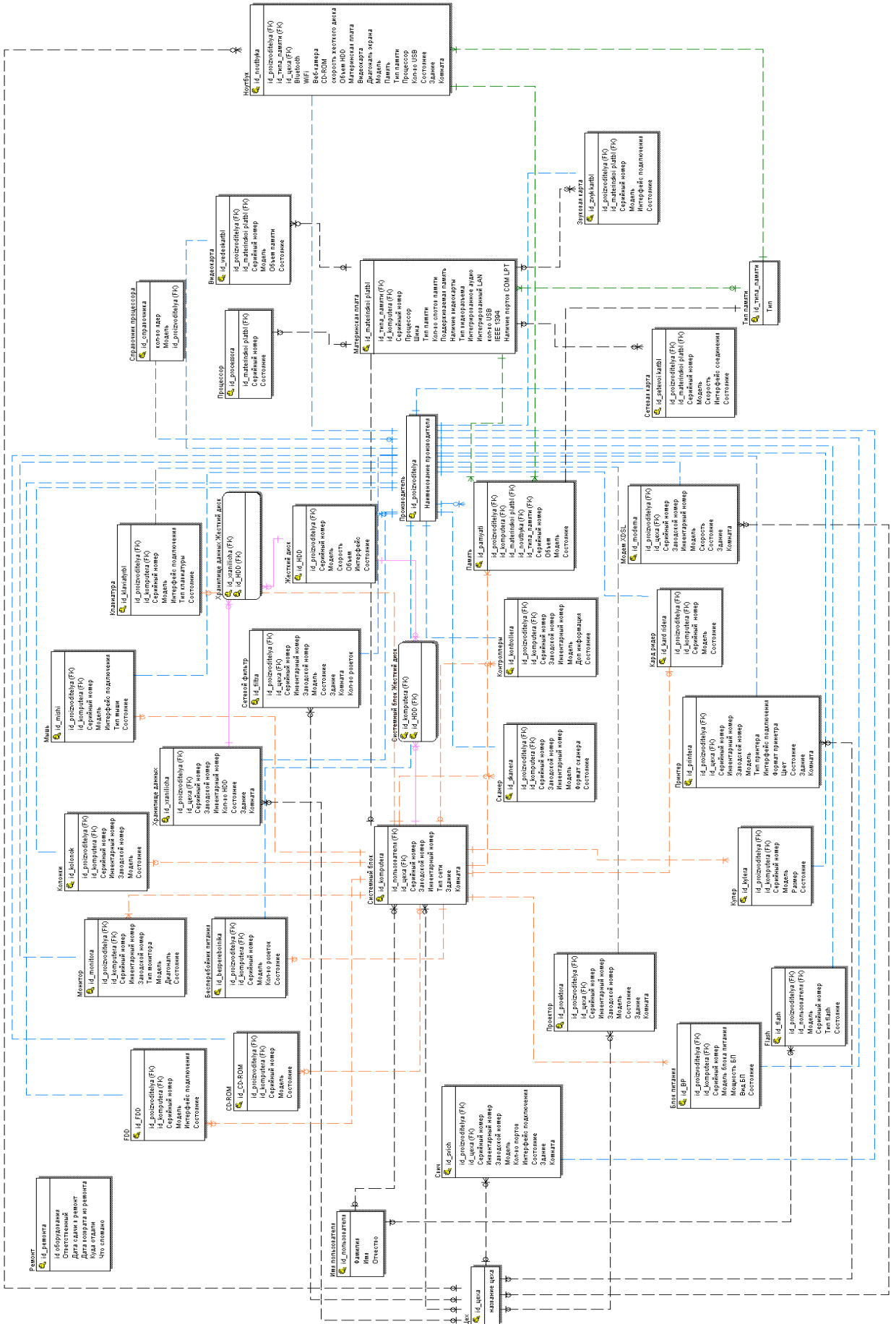


Рисунок А.1 — Логическая модель базы данных

Изм.	Лист	№ докум.	Подпись	Дата	

090301.2017.452 ПЗ

ПРИЛОЖЕНИЕ Б

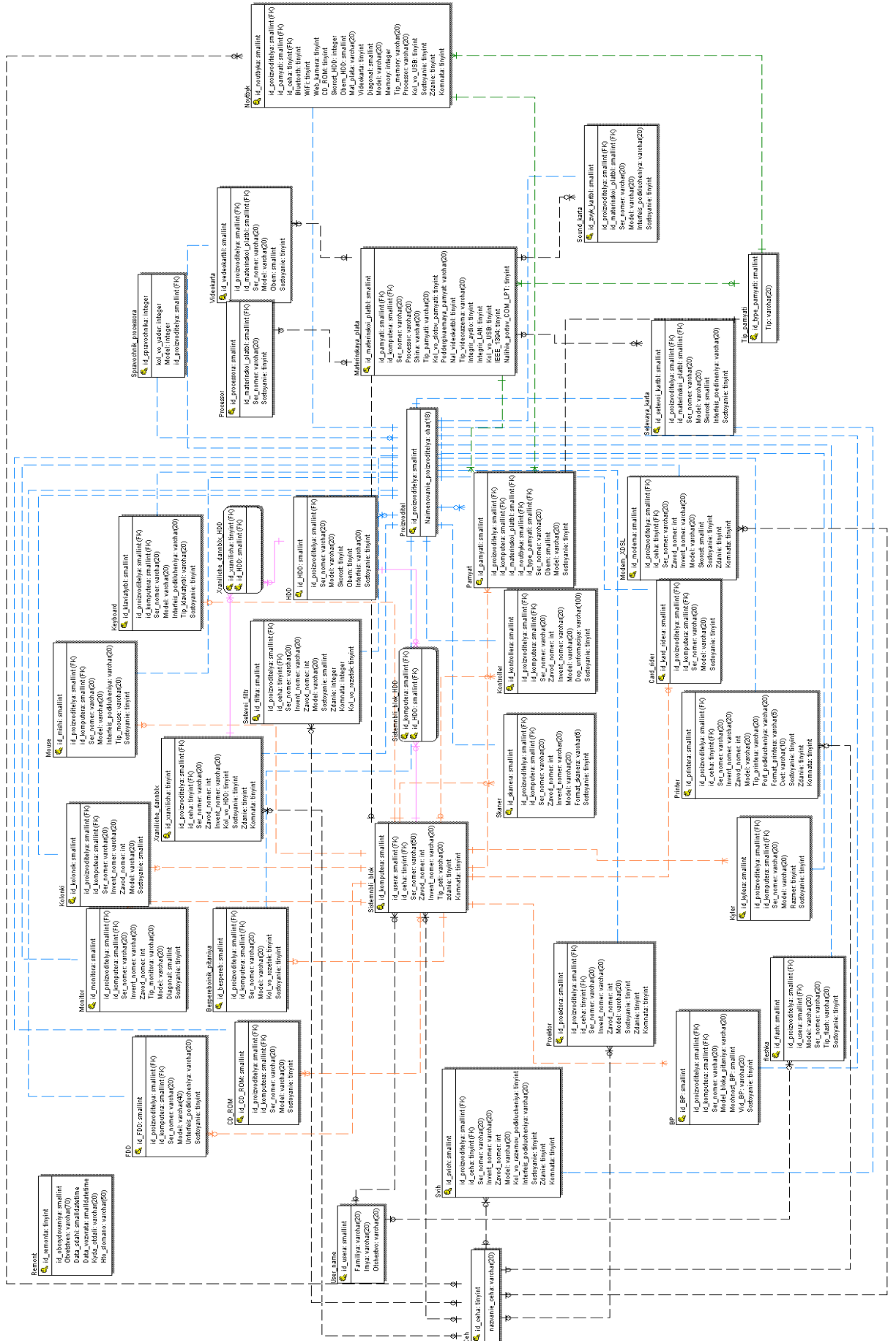


Рисунок Б.1 – Физическая модель базы данных