

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное учреждение
высшего образования
«Санкт-Петербургский политехнический университет Петра Великого»
(ФГАОУ ВО «СПбПУ»)

Институт среднего профессионального образования

Отделение информационных технологий

В. А. Андреев

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ВЫПОЛНЕНИЮ
КУРСОВОГО ПРОЕКТА
ПО МЕЖДИСЦИПЛИНАРНОМУ КУРСУ
ТЕХНОЛОГИИ РАЗРАБОТКИ И ЗАЩИТЫ БАЗ ДАННЫХ**

для специальностей:

**09.02.03 Программирование в компьютерных системах
09.02.07 Информационные системы и программирование**

очная/заочная форма обучения



Санкт-Петербург
2020

РАССМОТРЕНА:

предметной (цикловой) комиссией

Информационные системы

Протокол №1 от «30» августа 2019 г.

Председатель ПЦК

В.А. Андреев _____

подпись

УТВЕРЖДАЮ

Директор ИСПО

_____ А.А. Гусаков

«___» _____ 2019 г.

Рекомендована

Методическим советом ИСПО СПбПУ

Протокол №1 от «18» сентября 2019 г.

Зам. директора по УМР

Е.Г. Конакина _____

подпись

Методические рекомендации по дисциплине МДК 02.02 (МДК 11.01) Технологии разработки и защиты баз данных разработаны на основе учебного плана и рабочей программы ИСПО ФГАОУ ВО «Санкт-Петербургский политехнический университет Петра Великого» для специальностей 09.02.03 Программирование в КС и 09.02.07 Информационные системы и программирование.

Разработчик: Андреев В. А., преподаватель Института среднего профессионального образования.

Рецензенты:

СОДЕРЖАНИЕ

1 Цели и задачи курсового проектирования.....	4
2 Формулировка задания и его объем	5
3 Основное содержание курсового проекта по разделам.....	6
3.1 Задание на выполняемую работу	7
3.2 Перечень сокращений и обозначений	7
3.3 Введение	7
3.4 Аналитический обзор	7
3.5 Диаграмма прецедентов использования (в нотации UML)	7
3.6 Инфологическая модель.....	8
3.7 Даталогическая модель и связи между таблицами.....	9
3.8 Основные дисплейные фрагменты (интерфейсы)	11
3.9 Исключительные ситуации.....	12
3.10 Реализация добавления, удаления и обновления	12
3.11 Разработанные запросы	12
3.12 Обеспечение целостности БД	12
3.13 Разграничение прав доступа. Защита данных.	14
3.14 Разработка алгоритмов (блок-схемы) программных модулей	15
3.15 Справочная система.....	16
3.16 Тестирование приложения.....	16
3.17 Заключение.....	16
3.18 Список использованных источников.....	16
3.19 Приложения.....	17
4 Общие требования оформления курсового проекта.....	18
5 Рекомендации по организации работ над проектом и примерный календарный план его выполнения	22
6 Порядок защиты и ответственность студента за выполнение задания курсового проекта...	23
7 Оценка курсового проекта	24
8 Рекомендованная литература.....	25
8.1 Дополнительные источники.....	26
ПРИЛОЖЕНИЕ А Примерная тематика курсовых работ.....	27
ПРИЛОЖЕНИЕ Б Примеры оформления титульного листа и листа задания	29

1 Цели и задачи курсового проектирования

Междисциплинарный курс МДК 02.02 (МДК 11.01) «Технологии разработки и защиты баз данных» является дисциплиной, предусмотренной рабочими планами специальностей 09.02.03 «Программирование в компьютерных системах» и 09.02.07 «Информационные системы и программирование», в которых предусмотрен также и курсовой проект.

Основной целью курсового проектирования является:

- закрепление практических умений, полученных студентами на лабораторных занятиях, и получение дополнительных навыков обследования предметной области, концептуального, логического и физического проектирования базы данных, освоение средств поддержания целостности БД, формирования разнообразных запросов и т. д.
- знакомство и использование единой системы программной документации (ЕСПД): комплекса государственных стандартов, устанавливающих взаимосвязанные правила разработки, оформления и обращения программ и программной документации и обеспечивающих снижение трудоемкости и повышение эффективности разработки, сопровождения и эксплуатации программных изделий.

Задачами курсового проекта являются проектирование и создание прикладной программы (веб-приложения), обеспечивающей создание и ведение базы данных по выбранной предметной области. На примере этой программы должны быть показаны примеры реализации вопросов, перечисленных выше, и связанных с конкретной СУБД и операционной системой.

При формулировании заданий на курсовое проектирование учитывалась тесная связь дисциплины МДК 02.02 (МДК 11.01) «Технологии разработки и защиты баз данных» с другими дисциплинами.

Рекомендуемые операционная система и технические средства:

Приложения разрабатываются под операционную систему Microsoft Windows. Требования к ПК должны соответствовать требованиям к данной операционной системе.

Рекомендуемые СУБД – phpMyAdmin, MySQL, PostgreSQL, MD, eSQL, Microsoft SQL Server и т.д.

Рекомендуемые программные средства: MS Visual Studio, Notepad++, uKit, Coda2 и т.д.

По согласованию с руководителем проекта может быть использована другие операционная система, локальная реляционная СУБД или клиент–серверная архитектура и среда RAD.

2 Формулировка задания и его объем

Задание на курсовой проект по дисциплине МДК 02.02 (МДК 11.01) «Технологии разработки и защиты баз данных» одинаково по структуре для каждого студента. Пример бланка задания на курсовой проект представлен в приложении Б.

Из предлагаемого преподавателем списка выбрать предметную область. По желанию предметную область можно предложить самостоятельно. Примерная тематика курсовых работ представлена в приложении А.

Оформить и подписать у преподавателя задание на курсовой проект.

Задание сформулировано следующим образом:

1 Дать сравнительную характеристику популярных СУБД, оценить существующие технологии доступа к данным и обосновать выбор ПО для курсового проектирования.

2 Произвести аналитический обзор имеющихся на рынке аналогичных программных продуктов, оценить их интерфейс и сформировать требования к интерфейсу разрабатываемого приложения с учетом выбранного ПО.

3 Построить инфологическую модель выбранной предметной области, а именно предложить список сущностей и список атрибутов, описывающих их с расстановкой связей между сущностями (ER-диаграмма).

4 Реализовать даталогическое проектирование базы данных, на основе инфологической модели.

5 Разработать интерфейсы программы/веб-приложения по работе с базой и форму отчетов.

6 Разработать алгоритмы (блок-схемы) программных модулей.

7 Предусмотреть наличие исключительных ситуаций.

8 Проработать вопросы по обеспечению целостности БД и разграничение прав доступа пользователей (администратор, редактор, пользователь и т. д.). Защита данных.

9 Составить программу, реализующую поставленную задачу.

10 Проанализировать результаты выполнения разработанной программы / веб-приложения.

11 Подготовить «пояснительную записку» к курсовому проекту.

12 Разработать инструкцию (руководство) системному программисту – администратору информационной системы.

Разработанная информационная система должна содержать не менее 5 таблиц. Необходимо организовать выборку информации из разработанной базы данных, т.е. сформулировать не менее 5 запросов разных типов, реализуемых средствами выбранного программного средства, например, выборка данных из связанных таблиц и т. д.

Система безопасности должна предусматривать как минимум ввод имени и пароля пользователя для регистрации в программе.

Интерфейс приложения должен предусматривать наличие функций для администратора базы данных.

Разработанное приложение должно содержать как минимум одну отчетную форму.

3 Основное содержание курсового проекта по разделам

Основное содержание курсового проекта по дисциплине МДК 02.02 (МДК 11.01) «Технологии разработки и защиты баз данных» должно быть отражено в пояснительной записке, содержание которой включает обязательную информацию, которая содержится в документах, разрабатываемых на программный продукт (веб-приложение) по стандартам ЕСПД.

Последовательность выполнения курсового проекта также соответствует стандартным этапам разработки программ, заключающаяся в разработке эскизного, технического и рабочего проектов. Пояснительная записка при этом является обязательным документом каждой стадии разработки программного продукта.

Типовая структура пояснительной записки к курсовому проекту:

Титульный лист (1 страница)

Задание на выполняемую работу (1 страница)

СОДЕРЖАНИЕ (1-2 страницы)

ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ (1 страница)

ВВЕДЕНИЕ (1-2 страницы)

1 Общий раздел (8-12 страниц)

2 Специальный раздел (15-20 страниц)

2.1 Диаграмма прецедентов использования (в нотации UML) - отражающая требования к интерфейсу создаваемой программы

2.2 Инфологическая модель

2.3 Даталогическая модель и связи между таблицами

2.4 Проект основных дисплейных фрагментов (интерфейсы)

2.5 Исключительные ситуации

2.6 Реализация добавления, удаления и обновления

2.7 Разработанные запросы

2.8 Обеспечение целостности БД

2.9 Разграничение прав доступа. Защита данных

2.10 Разработка алгоритмов (блок-схемы) программных модулей

2.11 Справочная система

2.12 Тестирование приложения

ЗАКЛЮЧЕНИЕ (1-2 страницы)

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ (1-2 страницы)

ПРИЛОЖЕНИЕ А Характеристики программного и аппаратного обеспечения (1-2 страницы)

ПРИЛОЖЕНИЕ Б Руководство системного программиста (7-10 страниц)

ПРИЛОЖЕНИЕ В Листинг SQL-скрипта (около 3-5 страниц)

ПРИЛОЖЕНИЕ Г Листинг приложения (не более 10 страниц)

3.1 Задание на выполняемую работу

В приложении Б приведена типовая форма задания и список тем для выбора предметной области (приложение А). По желанию предметную область можно предложить самостоятельно.

3.2 Перечень сокращений и обозначений

В данном разделе приводится перечень использованных в тексте пояснительной записки сокращений и условных обозначений.

Здесь также необходимо привести перечень ключевых слов. Этот перечень ключевых слов должен включать от 5 до 15 слов или словосочетаний из текста пояснительной записки, которые в наибольшей мере характеризуют ее содержание и обеспечивают возможность информационного поиска. Ключевые слова приводятся в именительном падеже и печатаются строчными буквами в строку через запятые.

3.3 Введение

Во введении необходимо кратко пояснить предметную область, актуальность разработки, сформулировать основные цели и задачи курсового проектирования.

3.4 Аналитический обзор

В аналитическом обзоре необходимо подробно описать предметную область, прописать основные понятия и определения, привести краткий обзор популярных СУБД и анализ современных технологий и интерфейсов доступа к данным для обоснования выбора соответствующего программного обеспечения.

Здесь также приводится краткий обзор аналогичных программных продуктов, т. е. таких продуктов, в основе работы которых лежит СУБД, для формирования требований к интерфейсу создаваемого приложения и т. д.

3.5 Диаграмма прецедентов использования (в нотации UML)

Моделирование предметной области является одним из наиболее важных этапов работ при проектировании программных систем.

Основными задачами при моделировании предметной области являются описание бизнес-процессов. Это описание фактически является результатом изучения технологии выполнения поставленной задачи, подлежащей автоматизации. На основе описанной технологии определяются виды деятельности, которые следует автоматизировать (бизнес-требования к будущей программной системе).

Т.о. описание бизнес-процессов необходимо в процессе:

- проектирования пользовательского интерфейса системы;
- проектирования базы данных системы;

- формирования альбома выходных форм системы.

Диаграммы прецедентов или вариантов использования (в нотации UML) позволяют показать взаимодействия между вариантами использования и действующими лицами, отражая функциональные требования к разрабатываемой системе с точки зрения пользователя.

Вариант использования представляет собой последовательность действий, выполняемых системой в ответ на событие, инициируемое некоторым внешним объектом (действующим лицом). Вариант использования описывает типичное взаимодействие между пользователем и системой и отражает представление о поведении системы с точки зрения пользователя. В простейшем случае вариант использования определяется в процессе обсуждения с пользователем тех функций, которые он хотел бы реализовать, или целей, которые он преследует по отношению к разрабатываемой системе.

Цель построения диаграмм вариантов использования - это документирование функциональных требований к программе в самом общем виде, поэтому они должны быть предельно простыми.

Диаграмма вариантов использования является самым общим представлением функциональных требований к системе. Для последующего проектирования системы требуются более конкретные детали, которые описываются в других документах.

3.6 Инфологическая модель

Моделирование предметной области является одним из наиболее важных этапов работ при проектировании программных систем масштаба предприятия. Основными задачами при моделировании предметной области являются описание:

- бизнес-процессов предприятия;
- действующих лиц бизнес-процессов и их функций, подлежащих автоматизации в привязке к структуре автоматизируемого предприятия;
- бизнес-сущностей;
- сценариев выполнения бизнес-функций, подлежащих автоматизации.

Описание бизнес-процессов используются для отображения технологии выполнения производственной задачи, подлежащей автоматизации. На основе описанной технологии определяются виды деятельности, которые следует автоматизировать (бизнес-требования к будущей информационной системе).

Рассмотрев основные бизнес-процессы, характеризующие информационные потоки, можно переходить к стадии проектирования базы данных информационной системы.

В базе данных отображается какая-то часть реального мира. Естественно, что полнота ее отображения будет зависеть от целей создаваемой информационной системы. Для того чтобы база данных адекватно отражала предметную область, проектировщик базы данных должен хорошо представлять себе все нюансы, присущие данной предметной области, и уметь отобразить их в базе данных. Вполне, вероятно, что при рассмотрении бизнес-процессов, могут возникнуть сомнения, в том какие бизнес-сущности следует

автоматизировать и переносить в БД, а какие оставить так. Поэтому важно разобраться, как функционирует предметная область, прежде чем начинать проектирование базы данных.

Для формирования БД предметная область, подлежащая автоматизации, должна быть предварительно описана. Для этого в принципе может использоваться и естественный язык, но его применение имеет много недостатков, основными из них являются громоздкость описания и неоднозначность его трактовки. Поэтому обычно для этих целей используют искусственные формализованные языковые средства.

Т.о. цель инфологического моделирования – обеспечение наиболее естественных для человека способов сбора и представления той информации, которую предполагается хранить в создаваемой базе данных. В связи с этим под инфологической моделью БД понимают описание предметной области, выполненное с использованием специальных языковых средств, не зависящих от используемых в дальнейшем программных средств.

При построении инфологических моделей можно использовать язык **ER-диаграмм** (от англ. Entity-Relationship, т. е. сущность-связь). В них сущности изображаются помеченными прямоугольниками, ассоциации – помеченными ромбами или шестиугольниками, атрибуты – помеченными овалами, а связи между ними – ненаправленными ребрами, над которыми может проставляться степень связи (1 или буква, заменяющая слово "много") и необходимое пояснение связей.

3.7 Даталогическая модель и связи между таблицами

Рассмотрим процесс даталогического проектирования. Любая СУБД оперирует с допустимыми для нее логическими единицами данных, а также допускает использование определенных правил композиции логических структур более высокого уровня из составляющих информационных единиц более низкого уровня.

Кроме того, многие СУБД накладывают количественные и иные ограничения на структуру базы данных. Прежде чем приступить к построению даталогической модели, необходимо детально изучить особенности СУБД, определить факторы, влияющие на выбор проектного решения, ознакомиться с существующими методиками проектирования, а также провести анализ имеющихся средств автоматизации проектирования, возможности и целесообразности их использования.

Будучи математиком по образованию Э.Кодд предложил использовать для обработки данных аппарат теории множеств (объединение, пересечение, разность, декартово произведение). Он показал, что любое представление данных сводится к совокупности двумерных таблиц особого вида, известного в математике как **отношение** – relation. Наименьшая единица данных реляционной модели – это отдельное **атомарное** (неразложимое) для данной модели значение – **элемент данных**. Так, в одной предметной области фамилия, имя и отчество могут рассматриваться как единое значение, а в другой – как три различных значения.

Доменом называется множество атомарных значений одного и того же типа. Смысл доменов состоит в следующем. Если значения двух атрибутов берутся из одного и того же домена, то, вероятно, имеют смысл сравнения, использующие эти два атрибута (например, для организации транзитного рейса можно дать запрос "Выдать рейсы, в которых время вылета из Москвы в Сочи больше времени прибытия из Архангельска в Москву"). Если же значения двух атрибутов берутся из различных доменов, то их сравнение, вероятно, лишено смысла: стоит ли сравнивать номер рейса со стоимостью билета?

Отношение на доменах D_1, D_2, \dots, D_n (не обязательно, чтобы все они были различны) состоит из заголовка и тела. **Заголовок** состоит из такого фиксированного множества атрибутов A_1, A_2, \dots, A_n , что существует взаимно однозначное соответствие между этими атрибутами A_i и определяющими их доменами D_i ($i=1,2,\dots,n$). **Тело** состоит из меняющегося во времени множества **кортежей**, где каждый кортеж состоит в свою очередь из множества пар атрибут-значение $(A_i:V_i)$, ($i=1,2,\dots,n$), по одной такой паре для каждого атрибута A_i в заголовке.

Для любой заданной пары атрибут-значение $(A_i:V_i)$ V_i является значением из единственного домена D_i , который связан с атрибутом A_i .

Степень отношения – это число его атрибутов. Отношение степени один называют унарным, степени два – бинарным, степени три – тернарным, ..., а степени n – n -арным.

Кардинальное число или **мощность отношения** – это число его кортежей.

Поскольку отношение – это множество, а множества по определению не содержат совпадающих элементов, то никакие два кортежа отношения не могут быть дубликатами друг друга в любой произвольно-заданный момент времени. Пусть R – отношение с атрибутами A_1, A_2, \dots, A_n . Говорят, что множество атрибутов $K=(A_i, A_j, \dots, A_k)$ отношения R является возможным ключом R тогда и только тогда, когда удовлетворяются два независимых от времени условия:

1. Уникальность: в произвольный заданный момент времени никакие два различных кортежа R не имеют одного и того же значения для A_i, A_j, \dots, A_k .
2. Минимальность: ни один из атрибутов A_i, A_j, \dots, A_k не может быть исключен из K без нарушения уникальности.

Каждое отношение обладает хотя бы одним возможным ключом, поскольку по меньшей мере комбинация всех его атрибутов удовлетворяет условию уникальности. Один из возможных ключей (выбранный произвольным образом) принимается за его первичный ключ. Остальные возможные ключи, если они есть, называются альтернативными ключами.

Вышеупомянутые и некоторые другие математические понятия явились теоретической базой для создания реляционных СУБД, разработки соответствующих языковых средств и программных систем, обеспечивающих их высокую производительность, и создания основ теории проектирования баз данных.

Однако для массового пользователя реляционных СУБД можно с успехом использовать неформальные эквиваленты этих понятий:

- Отношение – Таблица (Сущность),

- Кортеж – Строка (Запись),
- Атрибут – Столбец, (Поле).

При этом принимается, что "запись" означает "экземпляр записи", а "поле" означает "имя и тип поля".

Реляционная база данных – это совокупность отношений, содержащих всю информацию, которая должна храниться в БД. Однако пользователи могут воспринимать такую базу данных как совокупность таблиц, построенных по инфологической модели, при этом:

- Каждая таблица состоит из однотипных строк и имеет уникальное имя.
- Строки имеют фиксированное число полей (столбцов) и значений (множественные поля и повторяющиеся группы недопустимы). Иначе говоря, в каждой позиции таблицы на пересечении строки и столбца всегда имеется в точности одно значение или ничего.
- Строки таблицы обязательно отличаются друг от друга хотя бы единственным значением, что позволяет однозначно идентифицировать любую строку такой таблицы.
- Столбцам таблицы однозначно присваиваются имена, и в каждом из них размещаются однородные значения данных (даты, фамилии, целые числа или денежные суммы).
- Полное информационное содержание базы данных представляется в виде явных значений данных и такой метод представления является единственным. В частности, не существует каких-либо специальных "связей" или указателей, соединяющих одну таблицу с другой.
- При выполнении операций с таблицей ее строки и столбцы можно обрабатывать в любом порядке безотносительно к их информационному содержанию. Этому способствует наличие имен таблиц и их столбцов, а также возможность выделения любой их строки или любого набора строк с указанными признаками.

Хотя даталогическое проектирование является проектированием логической структуры базы данных, на него оказывают влияние возможности физической организации данных, предоставляемые конкретной СУБД. Поэтому знание особенностей физической организации данных является полезным при проектировании логической структуры.

Логическая структура базы данных, а также сама заполненная данными база данных являются отображением реальной предметной области. Поэтому на выбор проектных решений самое непосредственное влияние оказывает специфика отображаемой предметной области, отраженная в инфологической модели.

3.8 Основные дисплейные фрагменты (интерфейсы)

Назначение данного раздела – проработка **в общем виде проблем**, связанных с вводом и выводом информации. Для отображения информации на экран дисплея необходимо разработать проект шаблона (интерфейсы). Шаблон может иметь вид совокупности

отдельных, выделенных зон, в которые м.б. помещены конкретные числовые данные, графики, диаграммы и т.д.

В этом разделе приводятся основные разработанные дисплейные фрагменты (интерфейсы) информационной системы и формы спроектированных отчетов.

3.9 Исключительные ситуации

Интерфейс, предупреждающий возникновение ошибок и проблем, должен информировать пользователя о статусе программы, давать ему контроль над работой приложения, общаться с пользователем во время ввода данных и т.д. Т.е. фактически закладываются возможности для нормального функционирования программы тем, что предусматривается программный анализ на наличие исключительных ситуаций.

Учитывая необходимость формирования инструкции администратору базы данных, необходимо предусмотреть соответствующие сообщения для него.

Описание предусмотренных реакций удобно в табличном виде, например:

Возникшая ситуация – Предусмотренная реакция

3.10 Реализация добавления, удаления и обновления

Ведение базы – это всегда отдельный компонент системы, в основе которой лежит СУБД. Качество проведенной нормализации проверяется именно этим разработанным сервисом в системе.

3.11 Разработанные запросы

В этой части записки необходимо привести все реализованные запросы и показать степень выполнения базового задания.

3.12 Обеспечение целостности БД

Целостность (от англ. integrity – нетронутость, неприкосновенность, сохранность, целостность) – понимается как правильность данных в любой момент времени. Но эта цель может быть достигнута лишь в определенных пределах: СУБД не может контролировать правильность каждого отдельного значения, вводимого в базу данных (хотя каждое значение можно проверить на правдоподобность).

Например, нельзя обнаружить, что вводимое значение 5 (представляющее номер дня недели) в действительности должно быть равно 3. С другой стороны, значение 9 явно будет ошибочным и СУБД должна его отвергнуть. Однако для этого ей следует сообщить, что номера дней недели должны принадлежать набору (1,2,3,4,5,6,7).

Поддержание целостности базы данных может рассматриваться как защита данных от неверных изменений или разрушений (не путать с незаконными изменениями и разрушениями, являющимися проблемой безопасности).

Современные СУБД имеют ряд средств для обеспечения поддержания целостности (так же, как и средств обеспечения поддержания безопасности).

Выделяют три группы правил целостности:

- Целостность по сущностям.
- Целостность по ссылкам.
- Целостность, определяемая пользователем.

Есть два правила целостности, общие для любых реляционных баз данных.

Не допускается, чтобы какой-либо атрибут, участвующий в первичном ключе, принимал неопределенное значение.

Значение внешнего ключа должно либо:

- быть равным значению первичного ключа цели;
- быть полностью неопределенным, т.е. каждое значение атрибута, участвующего во внешнем ключе должно быть неопределенным.

Для любой конкретной базы данных существует ряд дополнительных специфических правил, которые относятся к ней одной и определяются разработчиком.

Чаще всего контролируется:

- уникальность тех или иных атрибутов,
- диапазон значений (экзаменационная оценка от 2 до 5),
- принадлежность набору значений (пол "М" или "Ж").

Есть еще одно важное замечание, связанное с абсолютно различающимися требованиями к базе данных прикладных программистов и администратора базы данных. Первые хотели бы иметь в одном месте (например, в одной таблице) все данные, необходимые им для реализации запроса из прикладной программы или с терминала. Вторые же заботятся об исключении возможных искажений хранимых данных при вводе в базу данных новой информации и обновлении или удалении существующей. Для этого они удаляют из базы данных дубликаты и нежелательные функциональные связи между атрибутами, разбивая базу данных на множество маленьких таблиц.

Так как многолетний мировой опыт использования информационных систем, построенных на основе баз данных, показывает, что недостатки проекта невозможно устранить любыми ухищрениями в программах приложений, то опытные проектировщики не позволяют себе идти навстречу прикладным программистам (даже тогда, когда они сами являются таковыми).

Т. о. можно сформулировать основные советы:

- четко разграничивать такие понятия как запрос на данные и ведение данных (ввод, изменение и удаление);
- помнить, что, как правило, база данных является информационной основой не одного, а нескольких приложений, часть из которых появится в будущем;
- плохой проект базы данных не может быть исправлен с помощью любых (даже самых изощренных) приложений.

3.13 Разграничение прав доступа. Защита данных.

Преимуществом использования баз данных является то, что СУБД обеспечивает возможность более полной реализации принципа независимости прикладных программ от данных, чем это возможно при организации локальных файлов.

Наличие в составе СУБД средств, ориентированных на разные категории пользователей, делает возможной работу с базой данных не только профессионалов в области обработки данных, но и практически любого, причем это использование может быть как для их профессиональных целей, так и для удовлетворения потребности в информации в быту и т.п.

Остальные преимущества также становятся очевидными при детальном рассмотрении **основных требований, предъявляемых к СУБД:**

- адекватность отображения предметной области (полнота, целостность и непротиворечивость данных, актуальность информации, т. е. ее соответствие состоянию объекта на данный момент времени);
- возможность взаимодействия пользователей разных категорий и в разных режимах, обеспечение высокой эффективности доступа для разных приложений;
- дружелюбность интерфейсов и малое время на освоение системы, особенно для конечных пользователей;
- обеспечение секретности и конфиденциальности для некоторой части данных; определение групп пользователей и их полномочий;
- обеспечение взаимной независимости программ и данных;
- обеспечение надежности функционирования СУБД; защита данных от случайного и преднамеренного разрушения; возможность быстрого и полного восстановления данных в случае их разрушения; технологичность обработки данных, приемлемые характеристики функционирования СУБД (стоимость обработки, время реакции системы на запросы, требуемые машинные ресурсы и др.).

Вышеперечисленные моменты актуальны в любой задаче.

Также для любой системы актуальны хотя бы две группы пользователей: администратор базы данных и обычный пользователь.

Администраторы БД выполняют большой круг разнообразных функций, среди которых:

- Разработка процедур обеспечения целостности БД при вводе и корректировке данных, обеспечение ограничений целостности при параллельной работе пользователей в многопользовательском режиме;
- Первоначальная загрузка и ведение базы данных: разработка технологии первоначальной загрузки и ведения (изменения, добавления, удаления записей) БД, проектирование форм ввода, создание программных модулей, подготовка исходных данных, ввод и контроль ввода;

– Защита данных:

- 1) обеспечение парольного входа в систему: регистрация пользователей, назначение и изменение паролей;
- 2) обеспечение защиты конкретных данных: определение прав доступа групп пользователей и отдельных пользователей, определение допустимых операций над данными для отдельных пользователей, выбор/создание программно-технологических средств защиты данных;
- 3) Тестирование средств защиты данных;
- 4) Фиксация попыток несанкционированного доступа к информации;
- 5) Исследование возникающих случаев нарушения защиты данных и проведение мероприятий по их предотвращению;
- 6) Обеспечение восстановления БД: разработка программно-технологических средств восстановления БД, организация ведения системных журналов;
- 7) Анализ обращений пользователей к БД: сбор статистики обращений пользователей к БД, ее хранение и анализ (кто из пользователей, к какой информации, как часто обращался, какие выполнял операции, время выполнения запросов, анализ причин безуспешных, в том числе и аварийных, обращений к БД);
- 8) Анализ эффективности функционирования СУБД и развитие системы: анализ показателей функционирования системы (время обработки, объем памяти, стоимостные показатели), реорганизация и реструктуризация баз данных, изменение состава баз данных, развитие программных и технических средств;
- 9) Работа с пользователями: сбор информации об изменениях в предметной области, об оценке пользователями работы СУБД, определение регламента работы пользователей с СУБД, обучение пользователей, консультирование пользователей;
- 10) Организационно-методическая работа: выбор и создание методики проектирования БД, определение целей и направлений развития системы, планирование этапов развития СУБД, разработка и выпуск организационно-методических материалов.

3.14 Разработка алгоритмов (блок-схемы) программных модулей

В этом разделе приводятся пошаговые блок-схемы решения конкретной задачи (алгоритмы) с необходимой степенью детализации, отражающая функции и взаимосвязи отдельных элементов и компонент программы при решении поставленной задачи, т. е. фактически поясняется последовательность решения разрабатываемых модулей.

Этот раздел записки соответствует документу по ЕСПД ГОСТ 19.701-90 (ИСО 5807-85) Схемы алгоритмов, программ, данных и систем. Условные обозначения и правила выполнения.

3.15 Справочная система

Любое качественно сделанное приложение должно помимо своей функциональности обладать еще и возможностью помочь начинающему пользователю освоиться. Для этих естественных нужд пользователя и предусмотрены файлы справки. В этом разделе необходимо показать какими технологиями пользуется программист при их создании.

3.16 Тестирование приложения

Демонстрация хода решения конкретной задачи с использованием разработанной информационной системы с использованием копий экранов (скриншот). Демонстрация должна демонстрировать уровень проработки задания на курсовое проектирование.

В данном разделе разработчику также предоставляется возможность проанализировать ошибки, выявленные в ходе подготовки и выполнения задачи (м.б. ошибки из-за неправильной организации в среде операционной системы, машинные сбои и т. д.). Выявление причин нарушений позволит предложить ход работ по их устранению и даст возможность улучшить создаваемое приложение.

3.17 Заключение

В данном разделе необходимо сформулировать заключение по выполненной работе и полученных навыках.

Здесь также можно дать рекомендации по улучшению программы, т. к. анализ программы и результатов машинного эксперимента дает возможность выявить нерациональные приемы программирования и неудачные программные конструкции, ошибки в программе, ошибки используемых алгоритмов и постановки задачи, и указать путь исправления и улучшения программы.

Также необходимо привести сведения об используемой среде программирования и операционной системе, а также других программных продуктах, необходимых для решения поставленной задач, т.е. фактически в этом разделе необходимо перечислить все виды работ, выполненных с использованием ЭВМ.

3.18 Список использованных источников

Список используемой при выполнении курсового проекта литературы следует оформить в соответствии с ГОСТ 7.1–2003 (для примера см. список рекомендованной литературы). В этот список должны включаться только те книги, которые действительно используются при курсовом проектировании. Здесь также приводятся и ссылки на Internet источники.

3.19 Приложения

В данном разделе приводятся различные машинные документы, такие как листинги программы на исходном языке программирования, протоколы компоновки задачи с примером содержимого основных используемых файлов, копии экранов (скриншоты) и т. д.

Также производится обоснование выбора состава технических средств на основании проведенных расчетов и (или) анализа, распределение носителей данных, которые использует программа, требования к дополнительным устройствам, используемых при работе программы.

В качестве примера оформления документации по ЕСПД необходимо разработать "Руководство системного программиста- инструкцию администратору базы данных". Этот документ предусмотрен ГОСТ 19.503–79*. В нем приводятся все сведения, необходимые для установки и запуска программы с соответствующего носителя данных, объем программы, и, если необходимо, сведения об организации и предварительной подготовке исходных данных. Для данной инструкции важным будет пункт, связанный со средствами и сообщениями в системе, предназначенные специально для администратора базы данных.

Т. о. обязательными приложениями к курсовому проекту являются:

- Характеристики программного и аппаратного обеспечения;
- Руководство системного программиста – инструкции администратору базы данных;
- Листинг SQL-скрипта;
- Листинг программы.

Дополнительное содержание приложений к курсовому проекту определяется самим студентом. Сюда могут также включаться, например, демонстрирующие разработанный интерфейс и подтверждающие выполнение тестовых примеров, различные схемы и алгоритмы, и т. д.

4 Общие требования оформления курсового проекта

Пояснительная записка (ПЗ) к курсовому проекту должна выполняться с учетом требований ГОСТ 7.32–2017 “СИБИД. Отчет о НИР. Структура и правила оформления”. Эти требования сегодня предъявляются ко всем пояснительным запискам, научным отчетам и т. д.

Общий объем ПЗ не должен быть менее 40 страниц, но и не должен превышать 60 страниц.

ПЗ оформляется в соответствии с **ГОСТ 7.32-2017**

ПЗ должна быть отредактирована и распечатана компьютерным способом на одной стороне листа белой бумаги формата А4 (210x297 мм). Текст набирается в формате Microsoft Word (**Times New Roman**) через **1 интервал (с коэффициентом 1.5)**, кегль 14.

Текст следует печатать, соблюдая следующие размеры полей: **правое - 15 мм, левое - 30 мм, верхнее и нижнее - 20 мм.**

Абзацы в тексте начинают отступом, равным **12,5-15 мм.**

Разрешается использовать компьютерные возможности акцентирования внимания на определенных терминах, формулах, теоремах, применяя шрифты разной гарнитуры.

Шрифт должен быть четким, средней жирности, плотность текста - одинаковая по всему отчёту, лист должен быть заполнен текстом не менее, чем на 1/2 от всего объема.

Первыми страницами отчета являются титульные листы (ТЛ). **На титульных листах номер страницы не проставляется.**

После титульных листов следует СОДЕРЖАНИЕ, ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ, ВВЕДЕНИЕ и т.д., которым необходимо проставлять нумерацию страниц. Страницы следует нумеровать арабскими цифрами, соблюдая сквозную нумерацию. **Номер страницы проставляется в центре нижней части листа без знаков препинания.**

Наименования структурных элементов ПЗ «СОДЕРЖАНИЕ», «ПЕРЕЧЕНЬ СОКРАЩЕНИЙ И ОБОЗНАЧЕНИЙ», «ВВЕДЕНИЕ», «ЗАКЛЮЧЕНИЕ», «СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ», «ПРИЛОЖЕНИЕ» служат заголовками структурных элементов ПЗ. Заголовки структурных элементов следует располагать в середине строки без точки в конце и печатать прописными буквами, не подчеркивая, без номера. Каждый структурный элемент отчета следует начинать с нового листа (страницы).

Разделы, подразделы, пункты и подпункты следует нумеровать арабскими цифрами и записывать с абзацного отступа. Разделы должны иметь порядковую нумерацию в пределах всего текста, за исключением приложений.

Пример - 1, 2, 3 и т. д.

Пример - 1.1.1, 1.1.2, 1.1.3 и т. д.

После номера раздела, подраздела, пункта и подпункта в тексте точку не ставят. Разделы, подразделы должны иметь заголовки. Пункты, как правило, заголовков не имеют. Заголовки разделов, подразделов и пунктов следует печатать с абзацного отступа с прописной буквы без точки в конце, не подчеркивая. Если заголовок состоит из двух предложений, их разделяют точкой. В заголовке не допускается перенос слова на следующую строку, применение римских цифр, математических знаков и греческих букв.

Если раздел состоит из одного подраздела, то подраздел не нумеруется. Если подраздел состоит из одного пункта, то пункт не нумеруется.

Заголовки не могут содержать размерностей и сокращений.

Заголовки разделов отделяются от основного текста сверху на 24пт, снизу на 12пт как показано на рисунке 1.

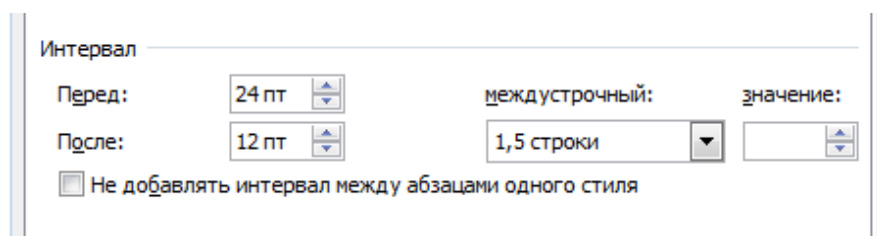


Рисунок 1 – Интервалы для заголовков

Иллюстрации (рисунки, чертежи, графики, схемы, диаграммы, фотоснимки, компьютерные распечатки, штриховые изображения) следует располагать в отчете непосредственно после текста, в котором они упоминаются впервые, или на следующей странице. На все иллюстрации должны быть даны ссылки в тексте.

Рисунки могут быть выполнены черными тушью или чернилами, карандашом на белой непрозрачной бумаге, компьютерным способом. Рисунки или фотографии размером менее формата А4 должны быть наклеены на листы формата А4. Допускается рисунки или фотографии маленького размера располагать на одном листе рядом друг с другом.

Иллюстрация должна иметь наименование, нумерацию и, при необходимости, пояснительные данные (подрисуночный текст). Слово "Рисунок" с номером и через тире - наименование рисунка, начинающееся с прописной буквы, помещают после пояснительных данных и располагают симметрично иллюстрации. Точка после наименования рисунка не ставится.

Пример:

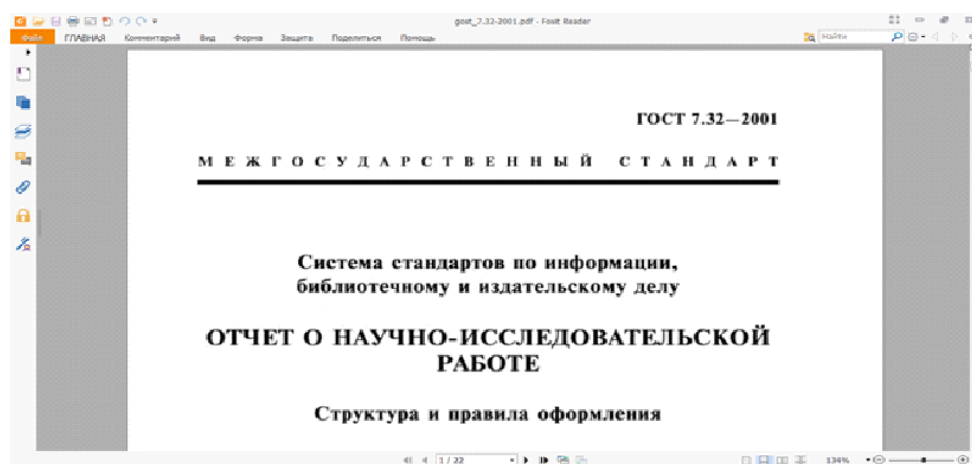


Рисунок 1 – ГОСТ 7.32 -2001

Иллюстрации, за исключением иллюстраций приложений, следует нумеровать последовательно арабскими цифрами сквозной нумерацией.

Иллюстрации, помещаемые в приложениях, обозначают отдельной нумерацией арабскими цифрами в пределах каждого приложения с добавлением перед каждой цифрой буквенного обозначения приложения (например, рисунок В.3).

Если в тексте одна иллюстрация, то она обозначается "Рисунок 1" или "Рисунок В.1", если иллюстрация приведена в приложении В.

При ссылках на иллюстрации следует писать, например, ". . . в соответствии с рисунком 1" или ". . . алгоритм функционирования системы управления (рисунок 1)".

Если иллюстрация не уместается по ширине листа, ее можно расположить вдоль длины листа, но таким образом, чтобы для ее рассматривания нужно было бы повернуть лист по часовой стрелке. Название таблицы должно отражать ее содержание, быть точным, кратким. Над таблицей без абзацного отступа помещается слово "Таблица" и ее номер, а через тире - название таблицы с прописной буквы.

Таблицы слева, справа и снизу, как правило, ограничивают линиями. Таблица выделяется сверху и снизу двумя интервалами.

Если в конце страницы таблица прерывается и будет продолжена на следующей странице, в первой части таблицы нижнюю горизонтальную линию не проводят. Головка таблицы должна быть отделена линией от остальной части таблицы.

Таблицу следует располагать в тексте после первой ссылки на нее. В ссылке следует писать слово таблица с указанием ее номера (например, ". . . результаты приведены в таблице 1" или ". . . были получены экспериментальные данные (таблица 1)").

Если после первого упоминания о таблице для ее размещения в тексте без переноса не хватает места, то таблицу лучше разместить на следующей странице.

Пример:

Таблица 2 – Название таблицы

Боковик	Головка				
	Аппаратура	Материал		Параметры	
	1 Реактор				
	2 Теплообменник				
	3 Смеситель				
	4 Приемная емкость				

Заголовки граф

Подзаголовки граф

Таблицы, за исключением таблиц приложений, следует нумеровать арабскими цифрами (без знака "№") сквозной нумерацией.

Таблицы, помещаемые в приложения, обозначают отдельной нумерацией арабскими цифрами в пределах каждого приложения с добавлением перед каждой цифрой буквенного обозначения приложения (например, Таблица В.1).

Если в тексте одна иллюстрация, то она обозначается "Рисунок 1" или "Рисунок В.1", если иллюстрация приведена в приложении В.

Делить головку таблицы по диагонали и включать в таблицу графу "Номер по порядку" не допускается. Нумерация граф таблицы арабскими цифрами допускается в тех случаях, когда в тексте имеется на них ссылка.

Если строки или графы таблицы выходят за формат листа, таблицу делят на части, которые переносят на другие листы или помещают на одном листе рядом, или одну под другой. Если части таблицы помещают рядом, то в каждой части повторяют головку, при размещении частей таблицы одна под другой - повторяют боковик.

При переносе части таблицы на другую страницу головку таблицы повторяют, а над ней без абзачного отступа пишут слова "Продолжение таблицы" с указанием ее номера. Название таблицы над ее продолжением не помещают. При переносе на другой лист части таблицы с большим количеством строк на новом листе повторяют ее головку; при переносе части таблицы с большим количеством граф повторяют ее боковик (ГОСТ 7.32 -91).

Пример:

Продолжение таблицы 2

Боковик	Головка				
	Аппаратура	Материал		Параметры	
	5 Брызгоотбойник				
	6 Перемешивающее устройство				

5 Рекомендации по организации работ над проектом и примерный календарный план его выполнения

Тема курсового проекта должна быть выдана не позднее, чем за два/два с половиной месяца до начала промежуточной аттестации и согласована с руководителем проекта. Задание на курсовой проект оформляется на типовом бланке (приложение Б.1).

Руководитель рекомендует список литературы, с которой студенту необходимо ознакомиться перед выполнением основной части курсового проекта. Студент имеет право дополнить предложенный список по своему усмотрению.

Работу над курсовым проектом по дисциплине МДК 02.02 (МДК 11.01) «Технологии разработки и защиты баз данных» рекомендуется выполнять в такой последовательности (наименование этапа работы номер недели):

Этапы работы	Сроки	Примечание
1 Получение задания на курсовое проектирование. Заполнение бланка задания.	1 неделя	Подписать задание у руководителя проекта
2 Аналитический обзор	2-3 неделя	Показать руководителю проекта
3 Проектирование инфологической и даталогической модели данных (схемы БД)	4 неделя	Утвердить правильность спроектированной схемы у руковод.
4 Проектирование интерфейсов	5 неделя	
5 Проектирование диаграмм прецедентов использования	6 неделя	
6 Программная реализация задачи	7-9 неделя	Показать руководителю проекта
7 Разработка алгоритмов программных модулей (блок-схемы)	10 неделя	
8 Создание справки	11 неделя	
9 Оформление пояснительной записки	2-12 неделя	Показать руководителю проекта
10 Сдача рабочей версии курсового проекта и пояснительной записки	12-13 неделя	
11 Защита курсового проекта	13-15 неделя	

6 Порядок защиты и ответственность студента за выполнение задания курсового проекта

В срок не менее чем за 7 дней до защиты руководителю проекта демонстрируется рабочая версия разрабатываемой в рамках курсового проекта программы и сдается на проверку Пояснительная записка. После проверки программы и записки руководитель либо допускает студента к защите, либо возвращает проект на доработку.

К защите студент также должен подготовить диск или внешний накопитель (флешка), содержащий все материалы по данному курсовому проекту, (в т. ч. и исходные тексты программ). Порядок защиты курсовых проектов и состав комиссии утверждается заведующей отделением. Студент делает доклад (5-7 минут), в котором кратко излагает результаты проектирования и демонстрирует разработанную программу.

Если студент на защите получает оценку "неудовлетворительно", то тема курсового проекта изменяется. Изменение темы курсового проекта происходит и в том случае, когда несколько пояснительных записок абсолютно одинаковы.

7 Оценка курсового проекта

Работа студента оценивается по пятибалльной шкале и состоит из результатов проверки основных этапов:

- Описание и анализ предметной области;
- Проектирование базы данных;
- Проектирование интерфейсов;
- Программная реализация;
- Разработка алгоритмов программных модулей;
- Оформление пояснительной записки;
- Доклад и защита курсового проекта.

При формировании итоговой оценки необходимо учитывать уровень самостоятельности студента и соблюдение установленного графика работы над курсовым проектом.

Оценкой *«отлично»* оцениваются курсовые проекты, выполненные в соответствии с заданием и вышеизложенными требованиями, выполненными самостоятельно, имеющими оригинальные технические решения, сданные в соответствии с графиком выполнения курсового проекта. При оценке проекта важную роль играют четкие ответы на поставленные вопросы. Повышает ценность курсового проекта его практическое использование на производстве или в учебном процессе.

Оценкой *«хорошо»* оцениваются курсовые проекты, имеющие частные недостатки в реализации проекта, некоторые пробелы в проработке отдельных вопросов, неполные ответы на вопросы, сданные в соответствии с графиком выполнения курсового проекта.

Оценкой *«удовлетворительно»* оцениваются курсовые проекты, имеющие существенные недостатки в реализации проекта, слабую проработку ключевых вопросов, недостаточно аргументированные ответы на вопросы, сданные с несущественным нарушением графика выполнения курсового проекта.

Оценкой *«неудовлетворительно»* оцениваются курсовые проекты, имеющие существенные недостатки в реализации проекта, очень плохую проработку ключевых вопросов, не аргументированные и неправильные ответы на вопросы, сданные с существенным нарушением графика выполнения курсового проекта.

8 Рекомендованная литература

- 1 Гвоздева, Т. В. Проектирование информационных систем: технология автоматизированного проектирования. Лабораторный практикум. Учебно-справочное пособие / Т. В. Гвоздева, Б. А. Баллод. - СПб. : Лань, 2018. - 156 с.
- 2 Коваленко, В. В. Проектирование информационных систем: Учебное пособие / В. В. Коваленко. - М. : Форум, 2015. - 976 с.
- 3 Перлова, О. Н. Проектирование и разработка информационных систем: Учебник / О. Н. Перлова. - М. : Академия, 2018. - 272 с.
- 4 Коннолли, Т. Базы данных. Проектирование, реализация и сопровождение. Теория и практика / Т. Коннолли. - М.: Вильямс И.Д., 2017. - 1440 с.
- 5 Вендеров, А.М. Проектирование программного обеспечения экономических и информационных систем: Учебник / А.М. Вендеров – М.: Финансы и статистика, 2013. – 544 с.
- 6 Мюллер, Р.Д. Проектирование баз данных и UML / Р.Д. Мюллер; Пер. с англ. Е.Н. Молодцова. - М.: Лори, 2013. - 420 с.
- 7 Дейтел, П. Android для программистов. Создаем приложения / П. Дейтел, Х. Дейтел, Э. Дейтел, М. Моргано. – СПб: Питер, 2013. – 1258 с.
- 8 Иванова, Г.С. Технология программирования: Учебник для вузов / Г.С. Иванова – М.: Изд-во МГТУ им. Н.Э. Баумана, 2013. – 320 с.
- 9 Липаев, В.В. Сертификация программных средств / В.В. Липаев. - М.: Синтег, 2010. – 348 с.
- 10 Лорен Дэрс, Шейн Кондер, Android за 24 часа. Программирование приложений под операционную систему Google, Рид Групп 2011
- 11 Майер, Р. Android 2. Программирование приложений для планшетных компьютеров и смартфонов / Р. Майер. – М.: Эксмо, 2013. – 644 с.
- 12 Хашими, С. Разработка приложений для Android / С. Хашими, С. Коматинени, Д. Маклин. – СПб: Питер, 2011. – 840 с.
- 13 Гаст, Х. Объектно-ориентированное проектирование: концепции и программный код / Х. Гаст. - М.: Диалектика, 2018. - 1040 с.

Электронные ресурсы:

- 14 Федеральное агентство по техническому регулированию и метрологии [Официальный сайт]. URL: <http://protect.gost.ru/>
- 15 Интуит. Качество ПО и методы его контроля [Официальный сайт]. URL: <http://www.intuit.ru/studies/courses/64/64/lecture/1874?page=1>
- 16 Техническая документация [Официальный сайт]. URL: <http://tdocs.su>
- 17 Карпов, В.Э. Об оформлении программной документации / В.Э. Карпов. [Электронный ресурс]. Режим доступа: <http://www.raai.org/about/persons/karpov/pages/ofdoc/ofdoc.html>

18 Пример составления руководства пользователя. [Электронный ресурс]. Режим доступа: <http://www.philosoft.ru/users-manual.zhhtml>

8.1 Дополнительные источники

- 1 Орлов, В.В. Технологии разработки программных продуктов. / В.В. Орлов. – СПб: Питер, 2009. – 437 с.
- 2 Малыхина, М.П. Базы данных: основы, проектирование, использование / М.П. Малыхина. - СПб.: BHV, 2007. - 528 с.
- 3 Баловсяк, Н. Видеосамоучитель создания реферата, курсовой, диплома на компьютере / Н. Баловсяк. – СПб: Питер, 2010. – 288 с.
- 4 Соловьев, И. В. Проектирование информационных систем. Фундаментальный курс / И.В. Соловьев, А. А. Майоров. - М. : Академический проект, 2009. - 398 с.
- 5 Соловьев, И. В. Проектирование информационных систем. Фундаментальный курс: Учебное пособие для высшей школы / И. В. Соловьев, А.А. Майоров; Под ред. В.П. Савиных. - М. : Академический проспект, 2009. - 398 с.
- 6 Федоров, Н.В. Проектирование информационных систем на основе современных CASE-технологий / Н. В. Федоров. - М. : МГИУ, 2008. - 280 с.
- 7 Пирогов, В. Информационные системы и базы данных: организация и проектирование: Учебное пособие / В. Пирогов. - СПб. : BHV, 2009. - 528 с.
- 8 Холл, М. Программирование для Web. Библиотека профессионала / М. Холл, Л. Браун – М. : Издательский дом "Вильямс", 2002. - 1264 с.

ПРИЛОЖЕНИЕ А

Примерная тематика курсовых работ

1. Информационная система книжного магазина
2. Информационная система гостиницы
3. Информационная система школы танцев
4. Информационная система проката велосипедов
5. Информационная система кафе
6. Информационная система сервиса ремонта ПК
7. Информационная система службы платных автопарковок
8. Информационная система компании морских перевозок
9. Информационная система частной поликлиники
10. Информационная система учета компрессорного оборудования
11. Информационная система всероссийской федерации волейбола
12. Информационная система службы аренды водного транспорта
13. Информационная система Кофейни
14. Информационная система продажи электроники
15. Информационная система магазина ножей
16. Информационная система интернет-провайдера
17. Информационная система кинотеатра
18. Информационная система проката гидроциклов
19. Информационная система школы...
20. Информационная система магазина художников
21. Информационная система для продажи квартир
22. Информационная система продуктового магазина
23. Информационная система отзывов на компании
24. Информационная система музея...
25. Информационная система электронных ЖД билетов
26. Информационная система туристического клуба
27. Информационная система продажа компьютерных игр
28. Информационная система картинной галереи
29. Информационная система парикмахерской
30. Информационная система библиотеки
31. Информационная система инвентарного учета
32. Информационная система курьерской службы
33. Информационная система компьютерного клуба
34. Информационная система студенческих театров
35. Информационная система игрового сервера
36. Информационная система автосервиса
37. Информационная система гостиницы

38. Информационная система детского лагеря
39. Информационная система интернет-магазина
40. Информационная система сервисов автопроката
41. Информационная система магазина игрушек
42. Информационная система мебельного магазина
43. Информационная система для определения плагиата
44. Информационная система торговой биржи
45. Информационная система учета сеансов психологов
46. Информационная система магазина автозапчастей
47. Информационная система компании ж/д перевозок
48. Информационная система службы доставки пиццы
49. Информационная система компании по выдаче займов
50. Информационная система Кафе
51. Информационная система сети гостиниц
52. Информационная система бюро по трудоустройству
53. Информационная система товарного склада
54. Информационная система финансовой биржи
55. Информационная система МВД
56. Информационная система кинотеатров.

Студенту разрешается выбрать предметную область самостоятельно (исключение составляют конкретные темы выполненных лабораторных работ по дисциплине МДК 02.02 (МДК 11.01) «Технологии разработки и защиты баз данных»), но только по согласованию с руководителем.

ПРИЛОЖЕНИЕ Б

Примеры оформления титульного листа и листа задания

Пример Б.1 Титульный лист

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное учреждение
высшего образования
«Санкт-Петербургский политехнический университет Петра Великого»
(ФГАОУ ВО «СПбПУ»)

Институт среднего профессионального образования

Отделение информационных технологий

Специальность: 09.02.03 Программирование в компьютерных системах

Дисциплина: МДК 02.02 Технологии разработки и защиты баз данных

Группа: 22928/*

КУРСОВОЙ ПРОЕКТ

на тему:

«Информационная система _____»

Студент(ка)

(ФИО)

(подпись)

Руководитель

(ФИО)

(подпись)

Оценка

(оценка)

(подпись)

(ФИО)

Санкт - Петербург
20XX

Пример Б.2 Задание на курсовое проектирование

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное учреждение
высшего образования
«Санкт-Петербургский политехнический университет Петра Великого»
(ФГАОУ ВО «СПБПУ»)

Институт среднего профессионального образования

Отделение информационных технологий

**Задание на курсовое проектирование
по МДК 02.02 «Технологии разработки и защиты баз данных»**

Специальность: 09.02.03 Программирование в компьютерных системах

студенту(ке) _____ группа _____
(фамилия, инициалы)

Тема: **Информационная система** *****

Исходные данные к проекту:

- 1 Литература по описанию объекта информатизации
- 2 Литература по инструментальным средствам разработки прикладных программ
- 3 Лекции, стандарты.
- 4 Интернет – сайты по предметной области

Перечень подлежащих разработке вопросов:

- 1 Аналитический обзор программных средств, технологий, аналогов ПП для создания базы данных и интерфейсов пользователей.
- 2 Проектирование реляционной базы данных (инфологическая и даталогическая модели с указанием связей (с пояснениями) между таблицами).
- 3 Разработка механизмов управления данными в БД (добавление, удаление и обновление данных).
- 4 Организация выборки информации из разработанной базы данных (сформулировать не менее 5 запросов всех типов, реализуемых средствами выбранной СУБД).
- 5 Проработка технологии доступа к базе данных (определение круга пользователей базы данных и оценка возможности разграничения полномочий пользователей).
- 6 Разработка ПО для решения поставленной задачи.
- 7 Разработка алгоритмов программных модулей.
- 8 Тестирование работы приложения.
- 9 Оформление документации (руководство системного программиста).
- 10 Оформление пояснительной записки.

Руководитель _____
(подпись, дата)

(инициалы, фамилия)

Задание принял к выполнению:

Студент _____
(подпись, дата)

(инициалы, фамилия)