



ДОНСКОЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
УПРАВЛЕНИЕ ДИСТАНЦИОННОГО ОБУЧЕНИЯ И ПОВЫШЕНИЯ  
КВАЛИФИКАЦИИ

Кафедра «Программное обеспечение вычислительной техники и  
автоматизированных систем»

**Методические указания**  
к курсовой работе  
по дисциплине  
**«Базы данных»**

Автор  
Кузин А.П.

Ростов-на-Дону, 2018

## Аннотация

Методические указания предназначены для студентов очной формы обучения направления 09.03.04 «Программная инженерия»

## Авторы

Ст.преп., ка. ПОВТиАС  
Кузин А.П.





## Оглавление

<b>ОБЩИЕ ПОЛОЖЕНИЯ .....</b>	<b>4</b>
<b>СТРУКТУРА ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ .....</b>	<b>5</b>
<b>ОФОРМЛЕНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ .....</b>	<b>7</b>
<b>ПРИМЕРНЫЕ ВАРИАНТЫ ЗАДАНИЙ К КУРСОВОЙ РАБОТЕ.....</b>	<b>9</b>

## 1. ОБЩИЕ ПОЛОЖЕНИЯ

Курсовая работа по предмету «Базы данных» является завершающим этапом изучения дисциплины. Она позволяет закрепить полученные в ходе изучения дисциплины навыки на практике, объединив в одной работе умения, вырабатываемые в ходе выполнения лабораторных работ, и теоретические знания, получаемые в ходе лекций.

В ходе выполнения курсовой работы студентом должно быть разработано программное обеспечение, функционал и требование к которому максимально приближены к реальным задачам, встречающимся в трудовой деятельности и связанных с созданием программ для мобильных устройств.

Программное средство может быть реализовано в любой среде программирования, используемой на кафедре при организации учебного процесса и подходящей для программирования мобильных устройств.

Выполнение курсовой работы может быть разбито студентом на следующие этапы:

1. **Анализ поставленной задачи и исследование инструментов**, необходимых для ее выполнения. Студентом должны быть рассмотрены необходимые для решения задачи возможности, технологии и средства программирования, необходимые для реализации работы.
2. **Общее проектирование системы**. Результатом данного этапа должны стать требования к функционалу создаваемого программного средства (ПС), пользовательскому интерфейсу, надежности, скоростным характеристикам (если таковые предъявляются), структурному строению и т.д. Желательно оформление таковых в виде соответствующих диаграмм языка UML. Данный этап включает также определение основных алгоритмов, необходимых при реализации проектируемого программного средства.
3. **Написание программного кода**. На данном этапе студент должен конкретизировать требования, выработанные на втором этапе и облачить их в структуры данных и функциональные блоки на выбранном языке программирования (ЯП). Код программы должен быть аккуратно оформлен: имена переменных должны соответствовать их назначению, соблюдены принятые в ЯП отступы, текст программы не должен быть «скомканным». Крайне желательно либо ручное форматирование кода, либо использование сторонних средств форматирования кода.
4. **Тестирование и отладка разработанного ПС**. На данном этапе студент должен разработать контрольные примеры и тесты, покрывающие все логические цепочки алгоритмов и охватывающие весь функционал программы, включая проверку поведения, созданного ПС в условиях неверных входных параметров. Найденные в ходе тестирования недостатки и ошибки должны быть устранены. Результатом данного этапа должно стать корректно работающее ПС. Желательно привести протокол испытаний программного средства, доказывающий его работоспособность и адекватность поставленной задаче.
5. **Написание пояснительной записки к курсовой работе**. Пояснительная записка должна в полной мере отражать суть, процесс, средства и результаты решения задачи, должна содержать все листинги и выходные документы.

## 2. СТРУКТУРА ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ

Правильно оформленная пояснительная записка является неотъемлемой частью курсовой работы. Она должна полностью описывать процесс создания студентом ПС, примененные при его создании технологии и полученный результат. Ниже приведен примерный план пояснительной записки с соответствующей нумерацией пунктов:

### ВВЕДЕНИЕ

Во введении необходимо показать актуальность решаемой задачи, провести анализ уже существующих программных решений, охарактеризовать существующие средства и технологии, позволяющие реализовать поставленную задачу.

### 1. АНАЛИЗ ПРЕДМЕТНОЙ ОБЛАСТИ

Данный пункт записки должен содержать результаты анализа задачи и выбора метода решения, подхода к проектированию программы и инструментария. Результатом анализа должна быть конкретная постановка задачи на курсовую работу, содержащая суть поставленной задачи, перечень входных и выходных данных разрабатываемого программного средства, а также требования к интерфейсу, языку и среде программирования.

### 2. ПРОЕКТИРОВАНИЕ БАЗЫ ДАННЫХ

#### 4.1 Описание внешней модели предметной области.

Должен включать описание представлений о предметной области не менее 2-х пользователей в следующем виде: описание объектов предметной области, их связей и связей между объектами; описание функциональных зависимостей, имеющих место в предметной области; описание способов, форм обработки и представления сведений о хранимой в базе данных информации;

#### 4.2 Модель предметной области

Должен быть представлен в виде схемы «Объекты связи».

#### 4.3 Схемы базовых отношений.

Должен содержать описание доменов всех атрибутов во всех отношениях.

#### 4.4 Множества функциональных зависимостей.

Описываются множества функциональных зависимостей всех отношений, а также для не менее чем 2-х отношений построить неприводимое множество функциональных зависимостей.

#### 4.5 Построение супер-ключа.

Построить один супер-ключ с формальным доказательством его уникальности и построение на его основе составного потенциального ключа с доказательством его избыточности. Обосновать выбор первичных ключей для всех спроектированных отношений;

#### 4.6 Нормализация отношений

Провести нормализацию до уровня не ниже ЗНФ при этом на каждом этапе нормализации требуется обосновать выбор первичных ключей для каждого отношения;

### 3. ПРОГРАММНОЕ КОНСТРУИРОВАНИЕ

Пункт в целом предполагает описание способа создания и кодирования программного средства, перечень используемых в ПС алгоритмов и структур данных и детальное описание ключевых алгоритмов программы.

#### 3.1 Проектирование интерфейса

В данном пункте должны быть приведены схематические эскизы пользовательского интерфейса с обоснованием выбранной модели интерфейса и конкретных деталей его реализации.

#### 3.2 Программное конструирование

В пункте должны быть описаны все разработанные модули, классы, методы классов и функции, реализованные студентом в ПС. Для методов классов и функций должны быть описаны их параметры и возвращаемые результаты.

Использование законченных участков кода и/или синтаксических конструкций языка программирования в данном пункте не допускается.

#### 3.3 Контрольные примеры и тесты

Приводится описание использованных тестов для проверки корректности разработанного ПС и один или несколько контрольных примеров, показывающих правильность функционирования ПС.

### 4. РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Руководство пользователя направлено на обучения конечных пользователей правилам работы с разработанным ПС. Руководство пользователя содержит, как правило, следующие части: аннотацию с кратким изложением документа; части с описанием того, как использовать ПС; пункт с описанием возможных проблем и способов их устранения. Также данный пункт должен содержать детальное описание процесса инсталляции программы на мобильное устройство пользователя.

### ЗАКЛЮЧЕНИЕ

Дать анализ проделанной работы, отметить достоинства созданного ПС, возможности его развития, практического применения. Отметить приобретенный практический опыт.

### СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

Список источников содержит ссылки на информационные источники, использованные студентом при разработке ПС. Список источников сортируется либо по алфавиту, либо согласно порядку появления ссылок на источники в тексте записки. Сами ссылки должны быть оформлены согласно ГОСТ Р 7.0.5-2008. В тексте пояс-

нительной записки должны содержаться ссылки на все перечисленные информационные источники.

## **ПРИЛОЖЕНИЕ А. ИСХОДНЫЕ КОДЫ ПРОГРАММНОГО СРЕДСТВА**

Приложение содержит основные исходные коды модулей, классов и функций ПС.

### **5. ОФОРМЛЕНИЕ ПОЯСНИТЕЛЬНОЙ ЗАПИСКИ**

Курсовая работа оформляется на одной стороне листа формата А4. Оформление работы выполняется в текстовом редакторе Microsoft Word версии не ниже 10 или редакторе OpenOffice.org Writer версии не ниже 3.0. Электронная версия документа должна быть сохранена в формате «\*.doc» или «\*.docx». В тексте не должно быть сокращений слов, за исключением общепринятых.

Текст работы следует писать на стандартных листах со следующими размерами полей: левое – не менее 30 мм, правое – не менее 10, нижнее – не менее – 20, верхнее – не менее 15. Общий объем должен быть в пределах 20-30 страниц печатного текста (без приложений). Работа сшивается в специальной папке или переплетается.

Основной шрифт абзаца – Times New Roman, 14 размер. Межстрочное расстояние – 1,5 интервала. Выравнивание по ширине страницы. Отступ первой строки абзаца – 1,25 см. При вставке в текст пояснительной записки участков кода на языках программирования допускается использование для них шрифта Courier New. В тексте пояснительной записки ЗАПРЕЩАЕТСЯ использование курсива или «жирного» шрифта.

Страницы работы следует нумеровать арабскими цифрами, соблюдая сквозную нумерацию по всему тексту, включая приложения. Титульный лист включается в общую нумерацию страниц, но номер страницы на титульном листе не проставляется.

В тексте пояснительной записки каждый раздел начинается с нового листа (это не относится к подразделам, пунктам, подпунктам и т.д.). Подразделы (имеющие сложные номера, например, 2.1 или 4.3) пишутся в продолжение страницы; название подраздела выделяется пустыми строками.

Заголовки разделов следует располагать в середине строки без точки в конце. Все буквы заголовков разделов – прописные. Подразделы, пункты и т.д. пишутся по правилам обычных предложений (первая буква – прописная) без точки в конце. Не допускается появление в тексте «висячих» заголовков, т.е. заголовков пунктов (подпунктов) текст которых начинается на другой странице, нежели сам заголовок.

Рисунки, иллюстрации, схемы и т.п. следует располагать сразу после текста, в котором они упоминаются впервые, или на следующей странице, если в указанном месте они не помещаются. На все иллюстрации должны быть ссылки в тексте. Иллюстрации должны иметь названия, которые помещают под иллюстрацией, их следует нумеровать арабскими цифрами сквозной нумерацией по всей работе. Допускается нумерация внутри раздела, в этом случае номер рисунка сложный. Иллюстрации обозначаются словом «Рисунок». Например, как показано на рис. 1.

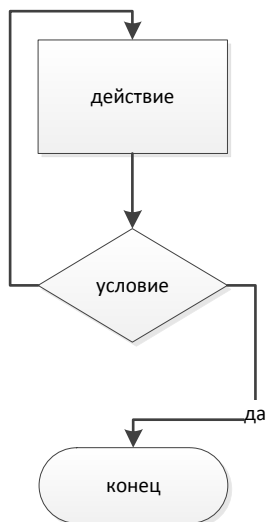


Рисунок 1 – Элемент блок-схемы

Таблицу следует располагать в работе непосредственно после текста, в котором она упоминается впервые, или на следующей странице. На все таблицы должны быть ссылки в тексте. Используется сквозная нумерация таблиц по всей работе. Номер таблицы размещается в правом верхнем углу таблицы после слова «Таблица». Каждая таблица должна иметь заголовок (наименование). Слово «Таблица» и заголовок начинаются с прописной буквы, точка в конце заголовка не ставится. Заголовки граф таблицы должны начинаться с прописных букв, подзаголовки – со строчных, если последние подчиняются заголовку. Заголовки граф указываются в единственном числе. Таблицу следует размещать так, чтобы читать ее без поворота работы. Если такое размещение невозможно, таблицу располагают так, чтобы ее можно было читать, поворачивая работу по часовой стрелке. Если цифровые или иные данные в какой-либо строке отсутствуют, то ставится прочерк.



## 6. ПРИМЕРНЫЕ ВАРИАНТЫ ЗАДАНИЙ К КУРСОВОЙ РАБОТЕ

В таблице 1 приведены примерные темы курсовых работ с пояснением требуемого функционала.

Таблица 1 – Примерные варианты тем курсовых работ

<b>№</b>	<b>Разработка приложения для работы с базой данной в выбранной предметной области</b>
1	Магазин одежды
2	Аэропорт
3	Парковка
4	Ресторан
5	Автомастерская
6	Магазин бытовой техники
7	Прокат автомобилей
8	Оружейный магазин
9	Музыкальный магазин
10	Почта
11	Воинская часть
12	Автосалон
13	Планировщик задач
14	Социальный рейтинг ДГТУ
15	Оператор сотовой связи