

Министерство образования и науки Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Московский государственный технический университет имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)



## **МЕТОДИЧЕСКИЕ УКАЗАНИЯ К НАПИСАНИЮ КУРСОВОЙ РАБОТЫ**

### **БАЗЫ ДАННЫХ**

*Москва, 2020 г.*

<b>Введение</b>	<b>4</b>
1. Общие требования	5
1.1. Сроки	5
1.2. Оценка курсовой работы	5
1.3. Тема курсовой работы	6
1.4. Требования к проектируемой базе данных	6
2. Содержание аналитического раздела	7
2.1 Общие вопросы проектирования информационных систем и программного обеспечения	7
2.1.1. Структурный анализ и проектирование	7
2.1.2. Объектно-ориентированный анализ и проектирование	9
2.2. Требования к аналитической части курсовой работы:	12
3. Содержание конструкторского раздела	13
3.2. Проектирование базы данных:	13
3.2. Требования к конструкторской части курсовой работы:	14
4. Содержание технологического раздела	15
4.1. Разработка интерфейса доступа к данным:	15
4.2. Требования к технологической части курсовой работы:	15
5. Содержание исследовательской части	16
6. Требования к содержанию структурных элементов РПЗ	17
6.1. Реферат	17
6.1.2. Перечень ключевых слов	17
6.2. Содержание	17
6.3. Введение	18
6.4. Заключение	18
6.5. Приложение	18
7. Правила оформления расчетно-пояснительной записки	19
7.1. Оформление текстового и графического материала	19
7.2. Оформление иллюстраций, таблиц и формул	19
7.3. Оформление текстов программ	20
7.4. Оформление списка использованных источников	20
7.5. Оформление приложений	21
8. Требования к содержанию структурных элементов презентации	22
8.1. Цель и задачи курсового проекта	22
8.2. Анализ существующих аналогов	22
8.3. ER-модель разработанной базы данных	22
8.4. Диаграмма вариантов использования	24
8.5. Результаты исследования	26
8.6. Заключение	26
8.7. Направления дальнейшего развития	27

Литература	28
Нормативно-правовые акты	28
Ссылки на печатные источники	28
Ссылки на электронные ресурсы	28
Материалы юридической практики	29
ПРИЛОЖЕНИЕ 1	30

## Введение

Цель курсовой работы - разработка и реализация информационной системы, прикладные программы которой используют данные, хранящиеся в базе данных, для автоматизации деятельности некоторого предприятия. В данном контексте слово «предприятие» используется как общий термин для достаточно независимой, коммерческой, научной, технической или другой организации. Предприятие может состоять из одного человека (с небольшой частной базой данных), быть крупной организацией (с очень большой базой данных) или представлять собой нечто среднее, например, промышленное предприятие, коммерческую организацию, учебное заведение, медицинское учреждение, библиотеку, банк или правительственное ведомство.

В рамках курсовой работы должны быть решены две основные задачи:

- проектирование и разработка базы данных
- разработка приложения (или приложений) доступа к базе данных.

Работа над курсовым проектом состоит из четырех частей, соответствующих четырем основным разделам РПЗ курсовой работы. Основные этапы работы над курсовой работой:

- аналитический раздел:
  - уточнение требований и формулировка ограничений предметной области;
  - анализ существующих решений;
- конструкторский раздел:
  - проектирование базы данных;
  - проектирование способов взаимодействия с базой данных;
- технологический раздел:
  - разработка базы данных и приложения доступа к базе данных;
  - тестирование разработанного приложения;
- исследовательский раздел:
  - проведение исследования разработанной базы данных.

## 1. Общие требования

### 1.1. Сроки

Основные этапы выполнения курсовой работы:

- 2 неделя - прием пожеланий по научным руководителям. Прием пожеланий организован с использованием Яндекс Формы. По окончании второй недели прием пожеланий завершается, формируется распределение студент-руководитель. Ссылка на форму - <https://forms.yandex.ru/cloud/63e7593ceb61465b0e8159a1>
- 3 неделя - прием тем курсового проекта/ прием ТЗ. Прием ТЗ организован с использованием Яндекс Формы. По окончании третьей недели прием пожеланий завершается, формируются списки студент-руководитель-тема. Если обратная связь от студента не получена, выдача руководителя и темы производится на усмотрение кафедры. Ссылка на форму - <https://forms.yandex.ru/cloud/63e7e62543f74f6e05ec29c9>
- 8 неделя - смотр (заочный) курсовых работ, включающий в себя проверку аналитического и конструкторского разделов. По результатам смотра студенты получают общий список замечаний к курсовым работам
- 13 неделя - смотр (заочный) курсовых работ, включающий в себя проверку технологического и исследовательского разделов. По результатам смотра студенты получают общий список замечаний к курсовым работам

Прием курсовых работ проводится:

- На 15 и 16 неделях семестра;
- В случае уважительных причин возможна организация дополнительных дней сдачи с предварительным уведомлением студентов на 17 неделе семестра.

### 1.2. Оценка курсовой работы

Для сдачи курсовой работы необходимо иметь следующий набор документов:

- Второй экземпляр ТЗ, подписанный научным руководителем и заведующим кафедрой.
  - В конце 4-ой недели студенты получают ТЗ в электронном виде. ТЗ формируется по данным, отправленным в форму для регистрации или выдается кафедрой в случае их отсутствия (Подача пожеланий по теме в форме “Курсовой проект по дисциплине Базы Данных” сданным ТЗ не считается);
  - Студент печатает ТЗ и подписывает его сам и у научного руководителя;
  - В конце 14 недели студент лично получает у ответственного за курсовую работу преподавателя экземпляр ТЗ, подписанный заведующим кафедрой, в обмен на ТЗ, подписанное самим студентом и научным руководителем. После чего подписывает ТЗ у научного руководителя (можно совместить с подписью РПЗ) и вшивает в РПЗ.
- Расчетно-пояснительная записка не менее 20 листов, подписанная научным руководителем. На титульном листе также должна присутствовать рекомендуемая научным руководителем оценка курсовой работы.
- Презентация к курсовой работе в двух экземплярах. Один экземпляр подшивается в РПЗ, второй сдается приемной комиссии. Для защиты курсовой работы также допустим электронный формат презентации.

- Разработанный программное обеспечение.
- Заключение о пройденном нормоконтроле.

Следующие пункты описывают возможные ситуации по снижению оценки:

- Несвоевременная сдача курсовой работы;
- Отсутствие или ненадлежащая реализация одной или нескольких частей курсового проекта.

### **1.3. Тема курсовой работы**

Тема курсовой работы обсуждается с назначенным руководителем. Тему в обязательном порядке необходимо утвердить у ответственного лица за курсовые работы по Бадам данных.

Если студент не успевает утвердить тему, тема выдается ответственным лицом за курсовые работы.

По истечении шести недель на кафедре вывешиваются окончательные списки тем курсовых работ. После чего тему работы можно сменить только заранее договорившись с руководителем и с ответственным лицом за курсовые работы.

### **1.4. Требования к проектируемой базе данных**

Обязательными требованиями к проектируемой базе данных являются:

- Наличие не менее 5 сущностей в базе данных;
- Наличие связей между сущностями базы данных (не менее одной связи многие-ко-многим);
- Наличие ролевой модели на уровне базы данных (не менее трех ролей);
- Наличие хранимой процедуры/функции/триггера на уровне базы данных;

## **2. Содержание аналитического раздела**

### **2.1 Общие вопросы проектирования информационных систем и программного обеспечения**

При проектировании информационных систем и программного обеспечения используются различные методологии, но двумя основными являются:

- структурный анализ и проектирование
- объектно-ориентированный анализ и проектирование.

#### **2.1.1. Структурный анализ и проектирование**

Основная идея структурного подхода заключается в декомпозиции исходной задачи на функции или процессы, которая приводит к созданию иерархии процессов и подпроцессов. Основные методы структурного подхода представлены стандартами семейства IDEF (<http://www.idef.com>).

В настоящий момент к семейству IDEF относятся следующие стандарты:

1. IDEF0 - методология функционального моделирования. С помощью наглядного графического языка IDEF0, проектируемая система представляется в виде набора взаимосвязанных функций (функциональных блоков). Как правило, моделирование средствами IDEF0 является первым этапом изучения любой системы.
2. IDEF1 – методология моделирования информационных потоков внутри системы, позволяющая отображать и анализировать их структуру и взаимосвязи.
3. IDEF1X (IDEF1 Extended) – методология построения реляционных структур. IDEF1X относится к типу методологий “Сущность-связь” (ER - Entity-Relationship) и, как правило, используется для моделирования реляционных баз данных.
4. IDEF2 – методология динамического моделирования развития систем. В настоящее время используются алгоритмы и их компьютерные реализации, позволяющие превращать набор статических диаграмм IDEF0 в динамические модели, построенные на базе “раскрашенных сетей Петри” (CPN – Color Petri Nets).
5. IDEF3 – методология документирования процессов, происходящих в системе, которая используется, например, при исследовании технологических процессов на предприятиях. С помощью IDEF3 описываются сценарий и последовательность операций для каждого процесса. IDEF3 имеет прямую взаимосвязь с методологией IDEF0 – каждая функция (функциональный блок) может быть представлена в виде отдельного процесса средствами IDEF3.
6. IDEF4 – методология построения объектно-ориентированных систем. Средства IDEF4 позволяют наглядно отображать структуру объектов и заложенные принципы их взаимодействия, тем самым, позволяя анализировать и оптимизировать сложные объектно-ориентированные системы.
7. IDEF5 – методология онтологического исследования сложных систем. С помощью методологии IDEF5 онтология системы может быть описана при помощи определенного словаря терминов и правил, на основании которых могут быть сформированы достоверные утверждения о состоянии рассматриваемой системы в некоторый момент времени. На основе этих утверждений формируются выводы о дальнейшем развитии системы, и производится её оптимизация.

В дополнение к IDEF0 для описания движения документов и обработки информации используются диаграммы потоков данных (DFD - Data Flow Diagrams). В отличие от IDEF0, где система рассматривается как взаимосвязанные работы, связи в DFD показывают лишь то, как объекты (включая данные) движутся от одной работы к другой. DFD отражает функциональные зависимости значений, вычисляемых в системе, включая входные значения, выходные значения и внутренние хранилища данных.

DFD содержит пять компонентов:

1. Процессы, которые преобразуют данные. Процессы самого нижнего уровня представляют собой функции без побочных эффектов (примером такой функции является вычисление комиссионного сбора за выполнение проводки с помощью банковской карточки). Весь граф потока данных тоже представляет собой процесс (высокого уровня). Процесс может иметь побочные эффекты, если он содержит нефункциональные компоненты, такие как хранилища данных или внешние объекты.
2. Потоки данных, которые переносят данные. Поток данных соединяет выход объекта (или процесса) с входом другого объекта (или процесса). Он представляет промежуточные данные вычислений.
3. Активные объекты, которые производят и потребляют данные. Активным называется объект, который обеспечивает движение данных, поставляя или потребляя их. Активные объекты обычно бывают присоединены к входам и выходам DFD.
4. Хранилища данных, которые пассивно хранят данные. Хранилище данных - это пассивный объект в составе DFD, в котором данные сохраняются для последующего доступа. Хранилище данных допускает доступ к хранимым в нем данным в порядке, отличном от того, в котором они были туда помещены. Агрегатные хранилища данных, как, например, списки и таблицы, обеспечивают доступ к данным в порядке их поступления, либо по ключам.
5. Потоки управления. DFD показывает все пути вычисления значений, но не показывает, в каком порядке значения вычисляются. Решения о порядке вычислений связаны с управлением программой, которое отражается в динамической модели. Эти решения, вырабатываемые специальными функциями, или предикатами, определяют, будет ли выполнен тот или иной процесс, но при этом не передают процессу никаких данных, так что их включение в функциональную модель необязательно. Тем не менее, иногда бывает полезно включать указанные предикаты в функциональную модель, чтобы в ней были отражены условия выполнения соответствующего процесса. Функция, принимающая решение о запуске процесса, будучи включенной в DFD, порождает в DFD поток управления.

Первым шагом при построении диаграмм потока данных является построение контекстных диаграмм. Обычно при проектировании относительно простых информационных систем строится единственная контекстная диаграмма со звездобразной топологией, в центре которой находится так называемый главный процесс, соединенный с приемниками и источниками информации, посредством которых с системой взаимодействуют пользователи и другие внешние системы. Для сложных информационных систем строится иерархия контекстных диаграмм. При этом контекстная диаграмма верхнего уровня содержит не главный единственный процесс, а набор подсистем, соединенных потоками данных. Контекстные диаграммы следующего уровня детализируют контекст и структуру подсистем.



### 2.1.2. Объектно-ориентированный анализ и проектирование

Основная идея объектно-ориентированного подхода состоит в рассмотрении предметной области и логического решения задачи с точки зрения объектов (понятий и сущностей). В процессе объектно-ориентированного анализа основное внимание уделяется определению и описанию объектов (или понятий) в терминах предметной области. В процессе объектно-ориентированного проектирования определяются логические программные объекты, которые будут реализованы средствами объектно-ориентированного языка программирования. Эти программные объекты включают в себя атрибуты и методы. И, наконец, в процессе конструирования или объектно-ориентированного программирования обеспечивается реализация разработанных компонентов и классов. Основные принципы объектно-ориентированного анализа и проектирования сводятся к следующему:

1. Анализ требований, во время которого выделяются основные процессы, происходящие в моделируемой системе и их формулировка в виде прецедентов. Прецедент – это текстовое описание процессов, происходящих в предметной области.
2. Объектно-ориентированный анализ предметной области. Задача этого шага в определении видов деятельности участников процесса и составлении концептуальной модели, которая отражает различные категории элементов предметной области. Причем не только виды деятельности участников, но и все относящиеся к делу понятия.
3. Объектно-ориентированное проектирование, при котором основное внимание сосредоточено на распределении обязанностей. Распределение обязанностей означает выделение задач и обязанностей различных программных объектов в приложении.

Наиболее важным моментом объектно-ориентированного анализа и проектирования является квалифицированное распределение обязанностей между компонентами программной системы. Обязанности объектов и их взаимодействия изображаются с использованием диаграмм классов и диаграмм взаимодействий. Основным инструментальным средством объектно-ориентированного анализа и проектирования является унифицированный язык моделирования UML (Unified Modeling Language). Технология UML, одобренная консорциумом OMG (Object Management Group), является мощным средством описания бизнес-процессов и представления их в той форме, которая устраивает как разработчиков, так и пользователей. В распоряжении проектировщика баз данных имеется множество UML-диаграмм (<http://www.uml.org>), которые часто используются на этапе определения требований, развертывания и на всех промежуточных этапах проектирования. Примеры различных UML-диаграмм приведены в Приложениях 5 и 6.

Хотя в основном UML применяется в объектно-ориентированном анализе и проектировании, однако этот язык моделирования может использоваться и во многих других типах анализа и моделирования систем, не требующих создания объектно-ориентированных приложений.

При выполнении проекта необходимо выбрать одну из методологий, и следовать ей на протяжении всей работы.

Приступая к работе, нужно добиться простого и ясного видения проблемы, при котором задачи и приоритеты проекта стали бы очевидными. Атрибутом хорошего видения проблемы является центральная идея (лейтмотив проекта).

Когда требования сформулированы, необходимо проанализировать их в целом и каждое по отдельности. Каждому требованию должен быть назначен тот или иной приоритет. Детализация приоритетов может быть любой, в зависимости от потребностей. Уровни приоритета можно определить следующим образом:

1. Необходимые. Эти требования обязательно должны быть воплощены в программе, без этого ее нельзя выпускать на защиту. Необходимые функции должны быть реализованы и испытаны как можно раньше.
2. Желательные. Присутствие этих требований крайне желательно. При достаточном обосновании от их реализации можно отказаться, но это не уменьшает важности их наличия в программе. Реализация и испытания желательных требований начинается сразу после обязательных.
3. Возможные. Эти требования также желательны, но реализуются в последнюю очередь и являются первыми кандидатурами на удаление, если вдруг возникнут проблемы со сроками.

Обычно проект начинается с создания документа, содержащего описание его основных целей. В описании формулируются очень абстрактные (и зачастую нетехнические) требования к системе, необходимые для понимания того, что же собой представляет система в целом. Пример такого документа приведен в Приложении 7.

В рамках любой предметной области действует широкий набор политик, законов и промышленных стандартов. Все эти контролирующие принципы в целом называются правилами делового регламента или деловыми правилами. Правила делового регламента - один из основных источников функциональных требований, поскольку они определяют возможности, которыми должна обладать система для выполнения правил.

Для создания и документирования правил делового регламента и их применения разработаны различные методики, а для организации правила делового регламента предложено множество разных таксономий (схем классификаций). Простейшая из них включает пять типов правил.

1. Факты (или инварианты) – это неизменные истины о сущности данных и их атрибутах. Зачастую они описывают связи и отношения между важными бизнес-терминами. Примеры фактов: стоимость билетов не возвращается, если покупатель изменяет маршрут; со стоимости доставки налог с продаж не берется.
2. Ограничения определяют, какие операции может выполнять система и ее пользователи. Вот некоторые слова и фразы, которые часто применяются при описании ограничивающего правила: должен, не должен, не может и только. Примеры ограничений:
  - a. постоянный посетитель библиотеки может отложить для себя до 10 книг;
  - b. экипажи коммерческих авиарейсов должны каждые 24 часа отдыхать не менее 8 часов;
3. Активаторы операций – это правила, при определенных условиях приводящие к выполнению каких-либо действий. Выразительным способом документирования активирующих операций являются таблицы решений. Примеры активаторов операций:
  - a. если срок хранения контейнера с химикатом истек, об этом необходимо уведомить лицо, у которого в данный момент находится контейнер;

- b. если клиент заказал книгу автора, написавшего несколько книг, клиенту следует предложить другие книги этого автора, прежде чем принять заказ,
- 4. Вывод - это правило, устанавливающее новые реалии на основе достоверности определенных условий. Вывод создает новый факт на основе других фактов или вычислений. Выводы зачастую записывают в формате «если — то», применяемом также при записи правил, активирующих операции; тем не менее, раздел «то» вывода заключает в себе факт или предположение, а не действие. Примеры выводов:
  - a. если платеж не поступил в течение 30 календарных дней с момента отправки счета, счет считается просроченным;
  - b. если поставщик не может поставить заказанный товар в течение пяти дней с момента получения заказа, заказ считается невыполненным;
- 5. Вычисления – это один из классов правил, которые определяют вычисления, выполняемые с использованием математических формул и алгоритмов. В отличие от активирующих операций правила вычислений можно рассматривать в качестве требований к программному обеспечению:
  - a. цена единицы товара снижается на 10% при заказе от 6 до 10 единиц, на 20% - при заказе от 11 до 20 единиц и на 30% - при заказе свыше 20 единиц;
  - b. общая стоимость заказа вычисляется как сумма стоимостей всех заказанных товаров, за вычетом скидок на количество, плюс государственные и местные налоги, действующие в округе, куда будет доставлен товар, плюс стоимость доставки и плюс необязательный страховой сбор.

Подробные функциональные и нефункциональные требования к разрабатываемой системе должны быть записаны в спецификации к требованиям. Существует несколько способов представления требований:

1. документация, в которой используется четко структурированный и аккуратно используемый естественный язык;
2. графические модели, иллюстрирующие процессы трансформации, состояния системы и их изменения, взаимодействия данных, а также логические потоки, классы объектов и отношения между ними;
3. формальные спецификации, где требования определены с помощью математически точных, формальных логических языков.

После формулирования требований проводят их анализ и осуществляют построение концептуальной и логической модели информационной системы, в соответствии с выбранной методологией. Подготавливается набор диаграмм, одной из которых, вероятно, будет ER-диаграмма (в случае структурного подхода).

## **2.2. Требования к аналитической части курсовой работы:**

- Анализ предметной области. Сравнительный анализ существующих “конкурентов”;
- Формулировка требований к разрабатываемой базе данных и проложению;
- Формализация и описание информации, подлежащей хранению в проектируемой базе данных;
- Проведенный анализ существующих баз данных на основе формализации данных;
- ER-диаграмма сущностей проектируемой базы данных в нотации Чена;
- Формализация и описание пользователей проектируемого приложения к базе данных;
- Диаграмма вариантов использования.

### 3. Содержание конструкторского раздела

В конструкторской части работы выполняется проектирование базы данных с учетом выбранной в аналитическом разделе БД.

#### 3.2. Проектирование базы данных:

В этой части осуществляются следующие мероприятия:

1. Реализуются все объекты базы данных: пользовательские типы данных или домены, таблицы, представления, индексы, умолчания, правила, ограничения целостности, хранимые процедуры, триггеры и функции. Создание перечисленных объектов необходимо оформить в виде сценария создания базы данных и разместить в приложении. Следует заметить, что триггеры как процедурные средства обеспечения целостности, хранимые процедуры, реализующие правила делового регламента, и функции, возвращающие табличные значения, трактуются как «объекты» базы данных наряду с таблицами, утверждениями, индексами и прочими структурами.
2. Реализуются запросы к базе данных, как на выборку, так и на обновление данных (часть запросов может быть запрограммирована в хранимых процедурах). Демонстрируется весь спектр запросов на выборку: простые, многотабличные, итоговые запросы с использованием статистических функций, запросы с группировкой, подчиненные и составные запросы.
3. Для обеспечения бесконфликтной работы в многопользовательском режиме при выполнении операций обновления составляются транзакции.
4. Описываются меры безопасности, находящиеся в компетенции СУБД. Принято различать шесть уровней системы безопасности БД:
  - a. физическую систему безопасности;
  - b. безопасность сетевого протокола;
  - c. доменную безопасность;
  - d. безопасность локального компьютера;
  - e. систему безопасности сервера БД;
  - f. безопасность приложений.

За реализацию первых четырех уровней системы безопасности БД отвечают сетевые и системные администраторы, за пятый уровень отвечают администраторы и разработчики БД, а за шестой - разработчики приложений.

В конструкторской части проекта основное внимание должно быть уделено системе безопасности сервера БД, занимающей пятый уровень модели безопасности.

Средства безопасности сервера БД можно разделить на четыре категории: аутентификация, авторизация, аудит и шифрование. Процесс предоставления доступа к БД состоит из двух фаз: сначала выполняется подключение к серверу БД (аутентификация), а затем открывается доступ к БД с ее объектами (авторизация). Разрешения на работу с объектами позволяют или запрещают пользователям выполнять действия над объектами БД, например таблицами и представлениями. Разрешения на выполнение SQL-операторов позволяют или запрещают пользователям создавать объекты, делать копии БД и файлов журнала. Действия, выполняемые в БД, отслеживаются с помощью аудита сервера БД.

Некоторые объекты БД, например, хранимые процедуры, разрешается зашифровать, чтобы защитить их содержимое. Для планирования системы безопасности необходимо определить требования к системе безопасности из списка системных требований, сформулированных в аналитической части проекта.

Для реализации системы безопасности следует определить пользователей, группы и роли, которым назначаются привилегии, и связать с ними требования, создав список соответствий «пользователь - действие». По возможности следует оптимизировать структуру системы безопасности с помощью вложенных групп и ролей, а также цепочек владения. Созданную систему безопасности необходимо испытать, подключаясь к БД с различными учетными именами, проверяя работу разрешений и выполняя аудит.

### **3.2. Требования к конструкторской части курсовой работы:**

- Диаграмма проектируемой базы данных;
- Описание сущностей проектируемой базы данных;
- Описание проектируемых ограничений целостности базы данных;
- Описание всех проектируемых процедур/функций/триггеров в формате схемы;
- Описание проектируемой ролевой модели на уровне базы данных (у каких ролей какой доступ и к каким объектам)

## **4. Содержание технологического раздела**

В технологической части работы выполняется разработка приложения и дается описание проведенного тестирования разработанного программного комплекса.

### **4.1. Разработка интерфейса доступа к данным:**

Программные интерфейсы для доступа к данным в значительной степени связаны с особенностями конкретной СУБД, тем не менее, в основе всех интерфейсов, применяемых в коммерческих СУБД, лежат одни и те же принципы. Например, СУБД MS SQL Server поддерживает такие интерфейсы доступа к данным: Embedded SQL for C (ESQL/C), SQL-DMO, DB-Library, ODBC, OLE DB, ADO, ADO.NET. Среди интерфейсов, поддерживаемых другими СУБД, следует отметить BDE - универсальный стандарт компании Inprise (SQL Links), OCI - интерфейс вызовов Oracle компании Oracle, JDBC - API для выполнения SQL запросов к базам данных из программ, написанных на языке Java компанией Sun Microsystems. При разработке приложения необходимо выполнить мотивированный выбор соответствующего интерфейса.

### **4.2. Требования к технологической части курсовой работы:**

- Обоснование выбора средств реализации базы данных и приложения (в том числе выбор СУБД);
- Описание сущностей реализованной базы данных;
- Описание реализованных ограничений целостности базы данных;
- Описание всех реализованных процедур/функций/триггеров в формате схемы;
- Описание ролевой модели на уровне базы данных;
- Описание методов тестирования разработанного функционала;
- Описание интерфейса доступа к базе данных.

## 5. Содержание исследовательской части

В качестве примеров исследований могут быть предложены следующие темы:

1. Методика аудита и оптимизация производительности сервера БД.
2. Оптимизация производительности сервера БД для полнотекстового поиска.
3. Сравнительный анализ методов доступа к метаданным: с помощью системных таблиц, системных хранимых процедур и представлений информационной схемы.
4. Исследование проблемы сеанса, остающегося открытым на стороне сервера после отключения пользователя.
5. Сравнительный анализ работы со строками: в хранимых процедурах, в компонентах промежуточного уровня или на клиенте.
6. Автоматическая проверка хранимых процедур на наличие нарушений стандартов кодирования.
7. Определение набора простых и составных индексов для базовых таблиц и материализованных представлений.
8. Исследование проблемы защиты строк подключения баз данных, паролей и других секретных параметров приложения.



## **6. Требования к содержанию структурных элементов РПЗ**

Структурными элементами расчетно-пояснительной записки являются:

- титульный лист;
- техническое задание на курсовую работу(ТЗ);
- реферат;
- содержание;
- введение;
- аналитический раздел;
- конструкторский раздел;
- технологический раздел;
- исследовательский раздел;
- заключение;
- список использованных источников;
- приложения.

### **6.1. Реферат**

Реферат должен содержать:

1. сведения об объеме РПЗ, количестве иллюстраций, таблиц, приложений, количестве использованных источников;
2. перечень ключевых слов;
3. текст реферата.

#### **6.1.2. Перечень ключевых слов**

Пример составления реферата приведен в ГОСТе 7.9-95. Перечень ключевых слов должен включать от 5 до 15 слов или словосочетаний из текста РПЗ, 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. Текст реферата должен отражать:

1. объект разработки (или исследования);
2. цель работы;
3. метод или методологию проведения работы;
4. результаты работы;
5. основные конструкторские и технологические характеристики разработанной информационной системы и программного обеспечения;
6. область применения;
7. прогнозные предположения о развитии объекта разработки.

### **6.2. Содержание**

Содержание включает введение, наименование всех разделов, подразделов, пунктов (если они имеют наименование), заключение, список использованных источников и наименование приложений с указанием номеров страниц, с которых начинаются эти элементы РПЗ.

### **6.3. Введение**

Во введении должны быть сформулированы цели и задачи проекта. Введение должно содержать оценку современного состояния решаемой проблемы, основание и исходные данные для разработки темы, обоснование необходимости проведения исследований, актуальность темы, связь данной работы с другими аналогичными работами.

### **6.4. Заключение**

Заключение должно содержать:

1. краткие выводы по результатам выполненной работы;
2. оценку полноты решений поставленных задач;
3. рекомендации по конкретному использованию результатов работы;
4. оценку научно-технического уровня выполненной работы в сравнении с лучшими достижениями в данной области.

### **6.5. Приложение**

В приложения рекомендуется включать материалы, связанные с выполненным проектом, которые по каким-либо причинам не могут быть включены в основную часть. В качестве приложения рекомендуется включать сценарий создания базы данных на языке SQL.

## **7. Правила оформления расчетно-пояснительной записки**

При оформлении расчетно-пояснительной записки следует придерживаться ГОСТ 7.32-2001 «Отчет о научно-исследовательской работе. Структура и правила оформления».

### **7.1. Оформление текстового и графического материала**

Расчетно-пояснительная записка должна быть выполнена печатным способом с использованием компьютера и принтера на одной стороне листа белой бумаги формата А4 через полтора интервала. Цвет шрифта должен быть черным, высота букв, цифр и других знаков - не менее 1,8 мм (кегель не менее 12). Текст записки следует печатать, соблюдая следующие размеры полей: правое - 10 мм, верхнее - 20 мм, левое и нижнее - 20 мм. Разрешается использовать компьютерные возможности акцентирования внимания на определенных терминах, формулах, теоремах, применяя шрифты разной гарнитуры.

Наименования структурных элементов записки «Реферат», «Содержание», «Введение», «Заключение», «Список использованных источников» служат заголовками структурных элементов записки.

Основную часть записки следует делить на разделы, подразделы и пункты. Разделы должны иметь порядковую нумерацию в пределах всего текста, за исключением приложений.

Страницы записки следует нумеровать арабскими цифрами, соблюдая сквозную нумерацию по всему тексту записки. Номер страницы проставляют в центре нижней части листа без точки. Титульный лист включают в общую нумерацию страниц записки. Номер страницы на титульном листе не проставляют. Иллюстрации и таблицы, расположенные на отдельных листах, включают в общую нумерацию страниц записки. Иллюстрации и таблицы на листе формата А3 учитывают как одну страницу.

### **7.2. Оформление иллюстраций, таблиц и формул**

Иллюстрации (чертежи, графики, схемы, компьютерные распечатки, диаграммы, фотоснимки) следует располагать в записке непосредственно после текста, в котором они упоминаются впервые, или на следующей странице. Иллюстрации могут быть в компьютерном исполнении, в том числе и цветные. На все иллюстрации должны быть даны ссылки в записке. Чертежи, графики, диаграммы, схемы, иллюстрации, помещаемые в записке, должны соответствовать требованиям государственных стандартов ЕСКД. Иллюстрации, за исключением иллюстрации приложений, следует нумеровать арабскими цифрами сквозной нумерацией. Если рисунок один, то он обозначается «Рисунок 1». Слово «рисунок» и его наименование располагают посередине строки. Допускается нумеровать иллюстрации в пределах раздела. В этом случае номер иллюстрации состоит из номера раздела и порядкового номера иллюстрации, разделенных точкой. Например, Рисунок 1.1. Иллюстрации, при необходимости, могут иметь наименование и пояснительные данные (подрисовочный текст). Слово «Рисунок» и наименование помещают после пояснительных данных, например: «Рисунок 1 - Диаграмма последовательностей «тонкого» Web-клиента». Иллюстрации каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения. Например, Рисунок А. 3. При ссылках на иллюстрации следует писать «... в соответствии с рисунком 2» при сквозной нумерации и «... в соответствии с рисунком 1.2» при нумерации в пределах раздела.

Таблицы применяют для лучшей наглядности и удобства сравнения показателей. Название таблицы, при его наличии, должно быть точным и кратким. Название таблицы следует помещать над таблицей слева, без абзацного отступа в одну строку с ее номером через тире, например: «Таблица 2.3 – Классы ADO.NET провайдера SQL Server». Таблицу следует располагать в записке непосредственно после текста, в котором она упоминается впервые, или на следующей странице. На все таблицы должны быть ссылки в записке. При ссылке следует писать слово «таблица» с указанием ее номера. Таблицу с большим количеством строк допускается переносить на другой лист (страницу). При переносе части таблицы на другой лист (страницу) слово «Таблица» и номер ее указывают один раз справа над первой частью таблицы, над другими частями пишут слово «Продолжение» и указывают номер таблицы, например: «Продолжение таблицы 1». При переносе таблицы на другой лист (страницу) заголовок помещают только над ее первой частью. Таблицу с большим количеством граф допускается делить на части и помещать одну часть под другой в пределах одной страницы. Если строки и графы таблицы выходят за формат страницы, то в первом случае в каждой части таблицы повторяется головка, во втором случае - боковик. Цифровой материал, как правило, оформляют в виде таблиц. Таблицы, за исключением таблиц приложений, следует нумеровать арабскими цифрами сквозной нумерацией. Допускается нумеровать таблицы в пределах раздела. В этом случае номер таблицы состоит из номера раздела и порядкового номера таблицы, разделенных точкой. Таблицы каждого приложения обозначают отдельной нумерацией арабскими цифрами с добавлением перед цифрой обозначения приложения. Если в документе одна таблица, то она должна быть обозначена «Таблица 1» или «Таблица В. 1», если она приведена в приложении В.

Уравнения и формулы следует выделять из текста в отдельную строку. Выше и ниже каждой формулы или уравнения должно быть оставлено не менее одной свободной строки. Пояснение значений символов и числовых коэффициентов следует приводить непосредственно под формулой в той же последовательности, в которой они даны в формуле. Формулы в записке следует нумеровать порядковой нумерацией в пределах всего отчета арабскими цифрами в круглых скобках в крайнем правом положении на строке. Формулы, помещаемые в приложениях, должны нумероваться отдельной нумерацией арабскими цифрами в пределах каждого приложения с добавлением перед каждой цифрой обозначения приложения, например формула (В. 1). Ссылки в тексте на порядковые номера формул дают в скобках. Пример - ... в формуле (1). Допускается нумерация формул в пределах раздела. В этом случае номер формулы состоит из номера раздела и порядкового номера формулы, разделенных точкой, например (3.1).

### **7.3. Оформление текстов программ**

Тексты программ должны оформляться в соответствии с «хорошим стилем» программирования, т.е. должны быть легко читаемы и хорошо документированы. В текстах должны быть комментарии. Комментарии не должны затенять структуру текста программы и быть ясными и краткими. Наименование программ и подпрограмм должны отражать их назначение. Логическая структура программы должна быть отражена в ее тексте.

### **7.4. Оформление списка использованных источников**

Список использованных источников следует оформлять в соответствии с требованиями ГОСТ 7.1–2003 «БИБЛИОГРАФИЧЕСКАЯ ЗАПИСЬ. БИБЛИОГРАФИЧЕСКОЕ ОПИСАНИЕ. Общие

требования и правила составления». При ссылке на использованный источник следует приводить порядковый номер источника, заключенный в квадратные скобки.

## **7.5. Оформление приложений**

Приложение оформляют как продолжение записки на последующих ее листах. В тексте записки на все приложения должны быть даны ссылки. Приложения располагают в порядке ссылок на них в тексте записки. Каждое приложение следует начинать с новой страницы с указанием наверху посередине страницы слова «Приложение». После слова «Приложение» следует буква, обозначающая его последовательность. Приложение должно иметь заголовок, который записывают симметрично относительно текста с прописной буквы отдельной строкой. Приложения обозначают заглавными буквами русского алфавита, начиная с А, за исключением букв Ё, З, И, О, Ч, Ъ, Ы, Ь. Допускается обозначение приложений буквами латинского алфавита, за исключением букв I и O. В случае полного использования букв русского и латинского алфавитов допускается обозначать приложения арабскими цифрами. Если в записке одно приложение, оно обозначается «Приложение А». Текст каждого приложения, при необходимости, может быть разделен на разделы, подразделы, пункты, подпункты, которые нумеруют в пределах каждого приложения. Перед номером ставится обозначение этого приложения. Приложения должны иметь общую с остальной частью документа сквозную нумерацию страниц.

## **8. Требования к содержанию структурных элементов презентации**

Структурными элементами презентации являются:

- титульный лист;
- цель и задачи курсового проекта;
- анализ существующих аналогов;
- ER-модель разработанной базы данных;
- диаграмма вариантов использования;
- схемы функций/процедур/триггеров;
- результаты исследования;
- заключение;
- направления дальнейшего развития.

### **8.1. Цель и задачи курсового проекта**

Второй слайд презентации должен включать в себя одну единственную цель и 4 задачи (по количеству разделов в курсовой работе) в формате:

- Проанализировать/Провести анализ ....;
- Спроектировать базу данных ....;
- Разработать базу данных и приложение ....;
- Исследовать зависимость чего-то от чего-то/Исследовать влияние чего-то на что-то.

Понятие «цель» означает результат, на достижение которого направлен определенный процесс. При написании курсовой работы предполагается, что студент рассчитывает получить какой-либо итог: получение теоретических выводов, практических результатов, разработка рекомендаций по развитию объекта исследования и т.д. Соответственно, целью не может быть само проведение исследования.

Для достижения желаемого результата студент осуществляет определенные действия, которые называются задачами. Другими словами, задачи – это способы достижения поставленной цели и этапы в продвижении к ней.

От того насколько четко сформулирована задача, зависит качество исследования. Задачи ставятся в виде утверждений, перечисляются по порядку в зависимости от сложности. Количество задач зависит от глубины исследования или сложности разработки и определяется студентом: в среднем от трех до семи задач.

### **8.2. Анализ существующих аналогов**

Анализ существующих аналогов включает в себя рассмотрение функциональных и нефункциональных требований к уже существующим решениям и разрабатываемой системе. Студентам необходимо обосновать необходимость нового решения в выбранной области.

### **8.3. ER-модель разработанной базы данных**

Наиболее известным представителем класса семантических моделей предметной области является модель «сущность-связь» или ER-модель, предложенная Питером Ченом в 1976 году. Модель сущность-связь — модель данных, позволяющая описывать концептуальные схемы

предметной области. Предметная область — часть реального мира, рассматриваемая в пределах данного контекста. Под контекстом здесь может пониматься, например, область исследования или область, которая является объектом некоторой деятельности. ER-модель используется при высокоуровневом (концептуальном) проектировании баз данных. С её помощью можно выделить ключевые сущности и обозначить связи, которые могут устанавливаться между этими сущностями. Во время проектирования баз данных происходит преобразование ER-модели в конкретную схему базы данных на основе выбранной модели данных (реляционной, объектной, сетевой или др.). ER-модель представляет собой формальную конструкцию, которая сама по себе не предписывает никаких графических средств её визуализации. В качестве стандартной графической нотации, с помощью которой можно визуализировать ERM, была предложена диаграмма сущность-связь (entity-relationship diagram, ERD). На практике понятия ER-модель и ER-диаграмма часто не различают, хотя для визуализации ER-моделей предложены и другие графические нотации. Основными понятиями ER-модели являются сущность, связь и атрибут (свойство).

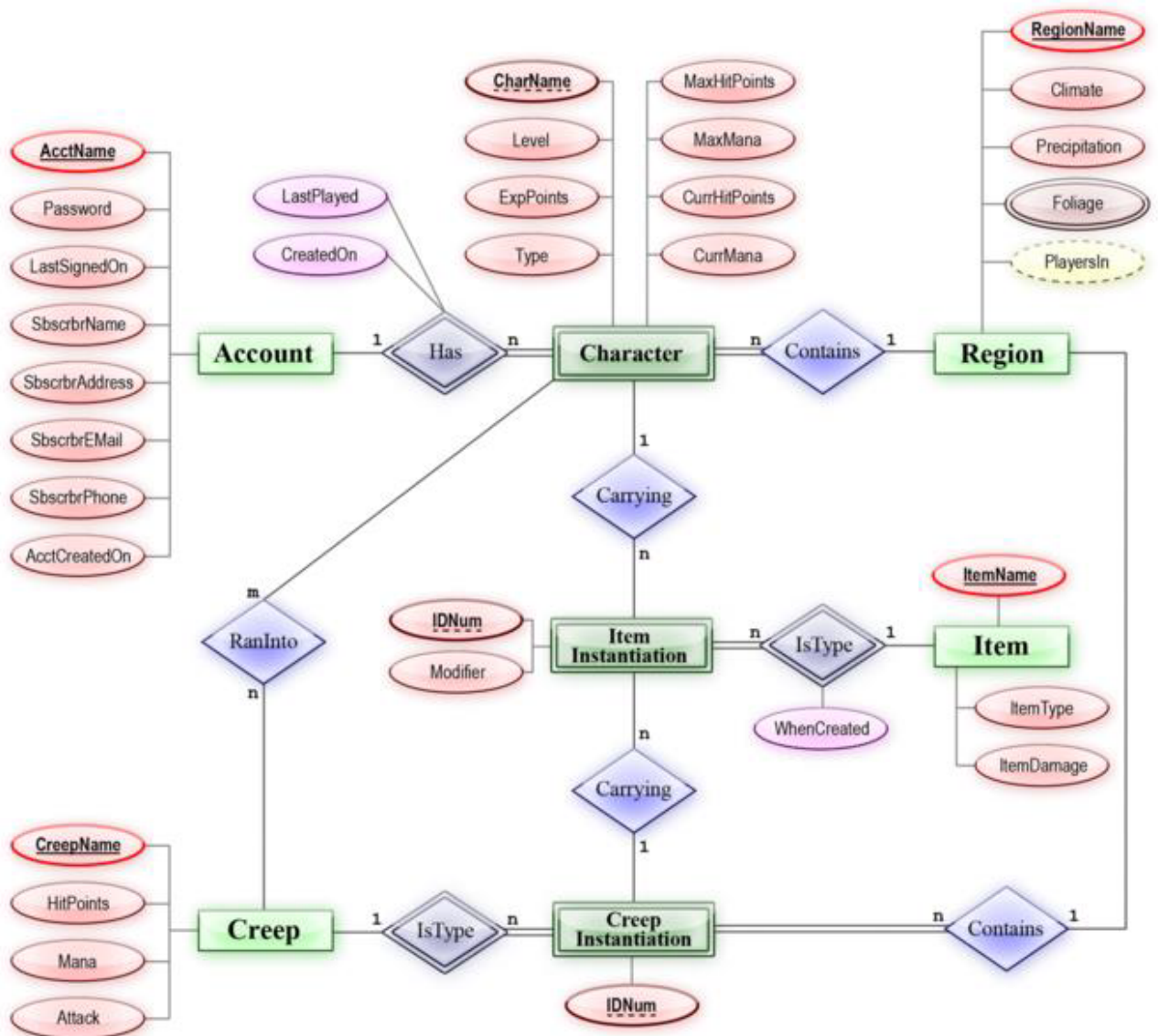
➤ Сущность - это реальный или представляемый объект, информация о котором должна сохраняться и быть доступна. В диаграммах ER-модели сущность представляется в виде прямоугольника, содержащего имя сущности. При этом имя сущности - это имя типа, а не некоторого конкретного экземпляра этого типа. Для большей выразительности и лучшего понимания имя сущности может сопровождаться примерами конкретных объектов этого типа. Каждый экземпляр сущности должен быть отличим от любого другого экземпляра той же сущности (это требование в некотором роде аналогично требованию отсутствия кортежей-дубликатов в реляционных таблицах). Сущности подразделяются на сильные и слабые. Сильные сущности существуют сами по себе, а существование слабых сущностей зависит от существования сильных.

➤ Связь - это ассоциация, устанавливаемая между сущностями. Эта ассоциация может существовать между разными сущностями или между сущностью и ей же самой (рекурсивная связь). Сущности, включенные в связь, называются ее участниками, а количество участников связи называется ее степенью. Участие сущности в связи может быть как полным, так и частичным. Связи в ER-модели могут иметь тип «один к одному», «один ко многим», «многие ко многим». Именно тип связи «многие ко многим» является единственным типом, представляющим истинную связь, поскольку это единственный тип связи, который требует для своего представления отдельного отношения. Связи типа «один к одному» и «один ко многим» всегда могут быть представлены с помощью механизма внешнего ключа, помещаемого в одно из отношений.

➤ Свойством сущности (и связи) является любая деталь, которая служит для уточнения, идентификации, классификации, числовой характеристики или выражения состояния сущности (или связи). Значения свойств каждого типа извлекаются из соответствующего множества значений, которое в реляционной терминологии называется доменом. Свойства могут быть простыми или составными, ключевыми, однозначными или многозначными, опущенными (т. е. «неизвестными» или «непредставленными»), базовыми или производными.

Более сложными элементами ER-модели являются подтипы и супертипы сущностей. Как в языках программирования с развитыми типовыми системами (например, в языках объектно-ориентированного программирования), вводится возможность наследования типа сущности, исходя из одного или нескольких супертипов.

На ER-диаграммах множества сущностей изображаются в виде прямоугольников, множества отношений изображаются в виде ромбов. Слабый тип сущности изображают в виде прямоугольника с двойным контуром. Слабый тип связи изображают в виде ромба с двойным контуром. Если сущность участвует в отношении, они связаны линией. Тип связи с частичным участием изображают двойной линией. Вид типа связи обозначается над линиями в виде соответствующих надписей возле типов сущностей. Например, если это вид бинарной связи «один ко многим», то делают надписи 1, n (или m), соответственно, возле соответствующих типов сущностей. Атрибуты изображаются в виде овалов и связываются линией с одним отношением или с одной сущностью. Именование сущности обычно выражается уникальным существительным, именование связи обычно выражается глаголом, именование атрибута обычно выражается существительным. Неизбыточный набор атрибутов, значения которых в совокупности являются уникальными для каждого экземпляра сущности, являются ключом сущности.



#### 8.4. Диаграмма вариантов использования

Диаграмма вариантов использования (диаграмма прецедентов или use-case диаграмма) описывает функциональное назначение системы. На диаграмме вариантов использования



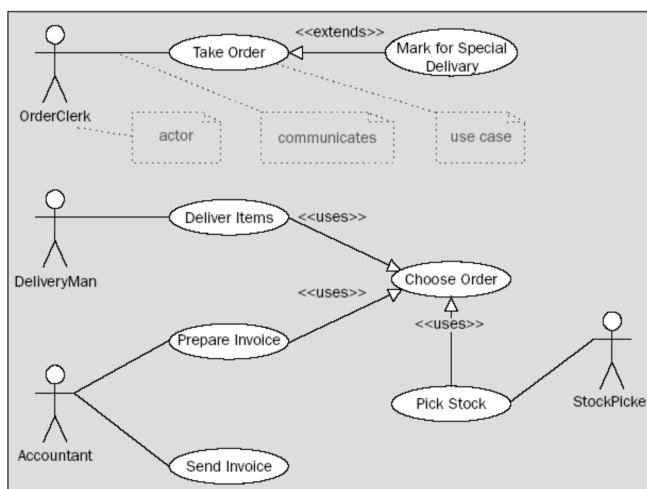
проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью вариантов использования.

Актер представляет собой любую внешнюю по отношению к моделируемой системе сущность, которая взаимодействует с системой и использует ее функциональные возможности для достижения определенных целей. Стандартным графическим обозначением актера является фигурка человечка, под которой записывается имя актера. Актеры взаимодействуют с системой посредством обмена сообщениями с вариантами использования.

Сообщение представляет собой запрос актером определенного сервиса системы и получение этого сервиса. Это взаимодействие может быть выражено посредством ассоциаций между отдельными актерами и вариантами использования. Кроме этого, с актерами могут быть связаны интерфейсы, которые определяют, каким образом другие элементы модели взаимодействуют с этими актерами.

Интерфейсы определяют совокупность операций, которые обеспечивают необходимый набор сервисов для актеров. Интерфейс изображается в виде маленького круга, рядом с которым записывается его имя. В качестве имени может быть существительное или строка текста. Если имя записывается на английском языке, то оно должно начинаться с заглавной буквы I. Графический символ отдельного интерфейса соединяется на диаграмме сплошной линией или пунктирной линией со стрелкой с тем вариантом использования, который его поддерживает. Сплошная линия указывает, что связанный с интерфейсом вариант использования должен реализовывать все необходимые для него сервисы. Пунктирная линия со стрелкой означает, что вариант использования предназначен для спецификации только того сервиса, который необходим для реализации данного интерфейса.

Вариант использования определяет законченный аспект или фрагмент поведения некоторой сущности без раскрытия её внутренней структуры. В качестве такой сущности может выступать система или любой элемент модели, который обладает собственным поведением. Каждый вариант использования соответствует отдельному сервису, который предоставляет моделируемая сущность по запросу актера, то есть определяет способ применения этой сущности. Вариант использования обозначается эллипсом, внутри которого содержится его краткое название или имя в форме глагола с пояснительными словами.



Для включения в модель произвольной текстовой информации используются примечания. На диаграмме примечания обозначаются прямоугольником с загнутым верхним правым углом. Внутри прямоугольника содержится текст примечания. Если в примечании указывается ключевое слово «constraint», то оно является ограничением, налагаемым на соответствующий элемент модели.

Между элементами диаграммы вариантов использования могут существовать различные отношения, которые описывают взаимодействие экземпляров актеров и вариантов использования:

1. ассоциации; отношение ассоциации специфицирует семантические особенности взаимодействия актеров и вариантов использования в графической модели системы, то есть, это отношение устанавливает, какую конкретную роль играет актер при взаимодействии с экземпляром варианта использования. Графически отношение ассоциации обозначается сплошной линией между актером и вариантом использования. Эта линия может иметь условные обозначения, такие как имя и кратность.
2. расширения; отношение расширения отмечает тот факт, что один из вариантов использования может присоединять к своему поведению некоторое дополнительное поведение, определенное для другого варианта использования. Графически отношение расширения обозначается пунктирной линией со стрелкой (вариант отношения зависимости), направленной от того варианта использования, который является расширением для исходного варианта использования. Данная линия со стрелкой помечается ключевым словом «extend».
3. обобщения; отношение обобщения служит для указания того факта, что некоторый вариант использования А может быть обобщен до варианта использования В. Графически Отношение обобщения обозначается сплошной линией со стрелкой в форме незакрашенного треугольника, которая указывает на родительский вариант использования.
4. включения; отношение включения между двумя вариантами использования указывает, что некоторое заданное поведение для одного варианта использования включается в качестве составного компонента в последовательность поведения другого варианта использования. Графически Отношение включения обозначается пунктирной линией со стрелкой, которая помечается ключевым словом «include» (включает).

## **8.5. Результаты исследования**

Результаты эксперимента должны отражать оценку и результаты личного эксперимента в рамках учебной исследовательской работы с описанием выводов.

## **8.6. Заключение**

Заключение курсовой работы должно отражать:

1. Выводы по теории и практической части исследования, которые были получены в ходе ученической исследовательской работы.
2. Подведение итога проделанной и практически готовой исследовательской работы, описание достигнутых целей и выполненных задач.

Выводы проделанной учащимся индивидуальной работы должны быть логично и последовательно связаны между собой, подкрепленными цифрами и фактами.

## **8.7. Направления дальнейшего развития**

В качестве направления дальнейшего развития студенту требуется внести предложения по устранению и решению выявленных в индивидуальной исследовательской работе проблем с описанием их практической значимости.

## Литература

По ГОСТ-2008 допускается не использовать тире между областями библиографического описания. Все ссылки должны быть оформлены единообразно: только с точкой, без тире между частями описания.

Сначала должна быть указа фамилия автора, потом его инициалы. При употреблении фамилии после слеш (/) – под. ред. или пер. – сначала пишутся инициалы, потом фамилия.

## Нормативно-правовые акты

Федеральный закон от 27.07.2006 N 149-ФЗ (ред. от 28.07.2012) «Об информации, информационных технологиях и о защите информации» // Собрание законодательства РФ, 31.07.2006. – N 31 (1 ч.). – Ст.3448

## Ссылки на печатные источники

1. Аболин Л.М. Психологические механизмы эмоциональной устойчивости человека. Казань: Изд-во Казан. ун-та, 1987 225 с. [Не «Казань: КГУ».]
2. Ажеж К. [Hagege С.] Человек говорящий: вклад лингвистики в гуманитарные науки: пер. с фр. Изд. 2-е, стер. М.: Едиториал УРСС, 2006 [Фамилия переводчика не указана; указание на перевод – после двоеточия; указание фамилии переводчика – см. пункт «Экман» ниже.]
3. Андреева Г.М. Социальная психология: учебник. 5-е изд., испр. и доп. М.: Аспект Пресс, 2006 [Не «Аспект-Пресс». Допускается не писать количество страниц в учебнике или книге.]
4. Барт Р. [Bart R.] Лингвистика текста: пер. с англ. // Новое в зарубежной лингвистике. М.: Прогресс, 1978 Вып. VIII: Лингвистика текста. С. 442–449. [Обязательно приводится оригинальное написание имени иностранного автора. Название выпуска можно опустить. Фамилия переводчика не указана; указание на перевод – после двоеточия; указание фамилии переводчика – см. пункт «Экман» ниже.]
5. Экман П. [Ekman P.] Психология лжи / пер. с англ. Н.Исуповой и др. СПб. и др.: Питер, 2008 [Фамилия переводчика указывается после косой черты, как и фамилии редакторов etc; указание на перевод без фамилии переводчика – см. выше.]
6. Campbell K.J., Collis K.F., Watsn J.M. Visual processing during mathematical problem solving // Educational Studies in Mathematics. 1995 Vol. 28(2). P. 177–194. [Недопустимо смешивать номер тома (Vol.), и номер выпуска (N). Настоятельно рекомендуем приведенный вариант описания: 28(2). Если выпуски/номера одного тома имеют сплошную нумерацию в пределах тома, допустимо указание только тома (Vol.). Пож., проверяйте описания на сайтах журналов. Символ & не используется. Кавычки в описаниях на латинице – только “лапки”, не «елочки».]

## Ссылки на электронные ресурсы

### С указанием DOI, по правилам описания печатных источников

Если электронной публикации присвоен цифровой идентификатор DOI, применяются правила описания печатных источников. В конце указывается DOI. Точка после DOI не ставится. В этом случае сетевой адрес (URL) не указывается.

С указанием материала [Электронный ресурс], для web-ресурсов – URL и даты обращения

В источниках на русском языке используются элементы: [Электронный ресурс], URL, (дата обращения: ...).

1. Белоус Н.А. Прагматическая реализация коммуникативных стратегий в конфликтном дискурсе [Электронный ресурс] // Мир лингвистики и коммуникации: электрон. научн. журн. 2006 N 4 URL: [http://www.tverlingua.by.ru/archive/005/5\\_3\\_1.htm](http://www.tverlingua.by.ru/archive/005/5_3_1.htm) (дата обращения: 15.12.2007).
2. Библиографическая ссылка. Общие требования и правила составления [Электронный ресурс]: издание официальное. М.: Стандартинформ, 2008 <http://protect.gost.ru/document.aspx?control=7&id=173511> (дата обращения: 05.10.2008).
3. Библиография по социальным и гуманитарным наукам, 1993-1995 [Электронный ресурс] / Ин-т науч. информ. по обществ. наукам (ИНИОН). М., 1995 1 электрон. опт. диск (CD-ROM). [Коллективный автор указан после косой черты]
4. Даль В.И. Толковый словарь живого великорусского языка Владимира Даля [Электронный ресурс]: подгот. По 2-му печ. изд. 1880–1882 гг. М.: АСТ и др.: 1998 1 электрон. опт. диск (CD-ROM).
5. Дрешер Ю.Н. Внедрение информационных технологий и системы управления качеством в библиотечно-информационное производств [Электронный ресурс] // Библиотеки информационные ресурсы в современном мире науки, культуры, образования и бизнеса: 12 междунар. конф. «Крым – 2005», Судак, 4–12 июня 2005 г. <http://www.gpntb.ru/win/inter-events/crimea2005/disk/163.pdf> (дата обращения: 24.12.2007).
6. Панасюк А.Ю. Имидж: определение центрального понятия в имиджелогии [Электронный ресурс] // Академия имиджелогии. 2004 26 марта. URL: [http://academim.org/art/pan1\\_2.html](http://academim.org/art/pan1_2.html) (дата обращения: 17.04.2008).
7. Парпалак Р. Общение в Интернете [Электронный ресурс] // Персональный сайт Романа Парпалака. 2006 10 декабря. URL: <http://written.ru> (дата обращения: 26.12.2006).
8. Соколова Е.Д., Березин Ф.Б., Барлас Т.В. Эмоциональный стресс: психологические механизмы, клинические проявления, психотерапия [Электронный ресурс] // Materia Medica. 1996 N 1(9). С. 5–25. URL: <http://flogiston.ru/library/sokolova-berezin> (дата обращения: 23.07.2008).

### **Материалы юридической практики**

Решение Арбитражного суда Новгородской области от 6 февраля 2013 года по делу №44-8209/2012. URL: <http://ras.arbitr.ru> (дата обращения 19.02.2013).

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ \_\_\_\_\_

КАФЕДРА \_\_\_\_\_

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**  
***К КУРСОВОЙ РАБОТЕ***  
***НА ТЕМУ:***

---

---

---

---

---

---

---

---

Студент _____	_____	_____
(Группа)	(Подпись, дата)	(И.О.Фамилия)

Руководитель курсовой работы	_____	_____
	(Подпись, дата)	(И.О.Фамилия)

Консультант	_____	_____
	(Подпись, дата)	(И.О.Фамилия)