

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Юдина Светлана Валентиновна

Должность: Директор АФ КНИТУ-КАИ

Дата подписания: 14.06.2019

Уникальный программный ключ:

ee380433c1f82e02d4d5ce32f117158c7c34ed0ff4b583f650075f51c9c70790

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное бюджетное образовательное
учреждение высшего профессионального образования Казанский национальный
исследовательский технический университет
им. А. Н. Туполева-КАИ
(КНИТУ-КАИ)
Альметьевский филиал**

**УПРАВЛЕНИЕ ИНФОРМАЦИОННЫМИ
СИСТЕМАМИ**

Методические указания к выполнению курсовой работы

Управление информационными системами: Методические указания к выполнению курсовой работы. – Альметьевск: Альметьевский филиал Казанского национального исследовательского технического университета им. А.Н. Туполева, 2019. – 42 с.

В методических указаниях приведены рекомендации и требования по структуре и содержанию основных тем курсовой работы, раскрыты основные направления, способствующие успешному выполнению работы.

Рецензенты:

Мокшин В.В. – кандидат технических наук, доцент кафедры ЕНДиИТ.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ.....	4
1. ОБЩИЕ ТРЕБОВАНИЯ К ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ .	5
1.1 УЧЕБНО-МЕТОДИЧЕСКИЕ ЗАДАЧИ КУРСОВОЙ РАБОТЫ И ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ.....	5
1.2 СОДЕРЖАНИЕ КУРСОВОЙ РАБОТЫ И ПОРЯДОК РАБОТЫ НАД ЕЕ РАЗДЕЛАМИ.....	6
2. ПОРЯДОК РАБОТЫ НАД КУРСОВОЙ РАБОТОЙ.....	8
3. РЕКОМЕНДАЦИИ ПО ВЫБОРУ ТЕМЫ КУРСОВОЙ РАБОТЫ.....	9
4. ОФОРМЛЕНИЕ КУРСОВОЙ РАБОТЫ	10
5. РЕКОМЕНДАЦИИ ПО РАБОТЕ С ЛИТЕРАТУРОЙ	18
6. ТЕМАТИКА КУРСОВЫХ РАБОТ.....	20
7. ИСХОДНЫЙ МАТЕРИАЛ ДЛЯ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ	22
8. ОСОБЫЕ РЕКОМЕНДАЦИИ К ВЫПОЛНЕНИЮ ОСНОВНОЙ ЧАСТИ КУРСОВОЙ РАБОТЫ	23
8.1 ОБЩИЕ ТРЕБОВАНИЯ	23
8.2 ОПИСАНИЕ ДИАГРАММЫ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ.....	24
8.3 ВАРИАНТ ИСПОЛЬЗОВАНИЯ	26
8.4 АКТЕРЫ	28
8.5 ИНТЕРФЕЙСЫ	30
8.6 ПРИМЕЧАНИЯ	31
8.7 ОТНОШЕНИЯ	32
8.8 ДИАГРАММА КООПЕРАЦИИ	35
8.9 ЗАКЛЮЧИТЕЛЬНЫЕ РЕКОМЕНДАЦИИ ПО РАЗРАБОТКЕ ДИАГРАММ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ	37
8.10 ЗАКЛЮЧИТЕЛЬНЫЕ РЕКОМЕНДАЦИИ ПО ПОСТРОЕНИЮ ДИАГРАММ КООПЕРАЦИИ	40
9. РЕКОМЕНДУЕМАЯ УЧЕБНАЯ, СПРАВОЧНАЯ И МЕТОДИЧЕСКАЯ ЛИТЕРАТУРА.....	42

ВВЕДЕНИЕ

Курсовая работа по управлению информационными системами – работа, после подготовки и защиты которой будущий бакалавр в области прикладной информатики должен получить возможность использовать полученные знания и навыки при выполнении дипломной работы. При выполнении настоящей курсовой работы студенты должны продемонстрировать свои способности в практической интеграции своих знаний основных дисциплин, необходимых при подготовке модели информационной системы. Курсовая работа по управлению информационными системами должна показать способность студента решать прикладные задачи в рамках реально существующей системы.

Задача настоящих методических указаний – дать студенту рекомендации по тематике, порядку подготовки, содержанию и оформлению курсовой работы, помочь ему правильно организовать работу над ней.

1. ОБЩИЕ ТРЕБОВАНИЯ К ВЫПОЛНЕНИЮ КУРСОВОЙ РАБОТЫ

1.1 УЧЕБНО-МЕТОДИЧЕСКИЕ ЗАДАЧИ КУРСОВОЙ РАБОТЫ И ТРЕБОВАНИЯ К КУРСОВОЙ РАБОТЕ

Курсовая работа является самостоятельной работой студента и позволяет преподавателям определить уровень знаний, полученных при изучении соответствующих дисциплин, предусмотренных учебным планом.

Основными учебно-методическими задачами курсовой работы являются:

1. Выявление глубины и основательности усвоения студентами знаний, в первую очередь, по управлению информационными системами, а также дисциплин, методы которых применяются в прикладной информатике.

2. Определение степени овладения навыками проведения научных и управленческих исследований и проектирования, умения применять выводы и положения науки к решению конкретных производственно-хозяйственных задач.

3. Выявление методических и организаторских способностей студента к реализации на практике разработанных им предложений.

Общая формулировка требований к курсовой работе может быть определена следующим образом:

1. Курсовая работа должна разрабатываться на конкретных производственных материалах, ее тема должна быть обусловлена реальными производственными задачами предприятия, на котором студент проходил практику или работает. Курсовая работа должна содержать конкретные разработки по управлению предприятием.

3. Курсовая работа должна носить комплексный характер, т.е. в процессе ее выполнения должны рассматриваться макроэкономические, научно-технические, маркетинговые, производственно-технологические, финансово-экономические и организационные стороны объекта проектирования.

4. Предложения и выводы курсовой работы должны иметь высокий уровень критериального обоснования выбора методов исследования и проектирования.

5. Курсовая работа должна содержать элементы новизны для предприятия в постановке задачи, методах их решения, критериальном обеспечении, методах исследования сложившейся производственной ситуации и проектирования управленческих изменений, организации реализации предложений курсовой работы, определении ее эффективности.

6. Объем курсовой работы может колебаться в пределах 30-50 страниц машинописного текста. Защита курсовой работы представляет собой доклад студента по итогам работы (не более 7 минут), ответы на вопросы комиссии по защите.

1.2 СОДЕРЖАНИЕ КУРСОВОЙ РАБОТЫ И ПОРЯДОК РАБОТЫ НАД ЕЕ РАЗДЕЛАМИ

Содержание и состав разделов курсовой работы определяются сложностью выбранной темы, особенностями объекта исследования, объемом и характером используемого материала. В любом случае курсовая работа должна содержать следующие основные разделы (табл. 1).

Таблица 1

Структура курсовой работы

Наименование и объем разделов	Краткое содержание разделов
Введение, 1 – 2 стр.	Постановка и формулирование целей и задач курсовой работы
1. Резюме объекта, 2 – 3 стр.	Содержание данных разделов курсовой работы представлено в соответствующем разделе настоящего пособия. В целом основные разделы работы должны дать максимально полное представление о моделировании программного обеспечения информационной системы.
2. Описание объекта и отрасли, 2 – 3 стр.	
3. Характеристика использования объекта, 2 – 3 стр.	
4. Исследование и анализ аналогов объекта, 2 – 3 стр.	
5. Конкуренция и конкурентное преимущество, 2 – 3 стр.	
6. Варианты использования, 2 – 3 стр.	
7. Диаграммы и описание вариантов использования, 2 – 3 стр.	
8. Описание интерфейсов (актеров), 2 – 3 стр.	
9. Кооперативные диаграммы и их описание, 5 – 8 стр.	
10. Дополнительные специфические моменты (зависит от темы), 1 – 2 стр.	
Заключение, 1 – 2 стр.	Краткое описание проделанной работы. Выводы и рекомендации по результатам работы.

Содержание разделов курсовой работы имеет свою специфику, однако общие рекомендации, которые могут быть использованы независимо от избранной темы, сводятся к следующим моментам.

Введение. Введение должно содержать краткую характеристику избранной темы, обоснование её актуальности, важности, рассматриваемых в курсовой работе вопросов для повышения эффективности деятельности предприятия. Студент должен аргументировать выбор объекта исследования, сформулировать цель курсовой работы и основные задачи, решения которых предусматриваются в курсовой работе. При необходимости там же указываются ограничения и условности представляемого проекта в связи с его учебным характером.

Заключение. Дается краткий итог проведенного проектирования. В сжатом виде формулируются важнейшие выводы, к которым пришел студент в результате проделанной работы, систематизируются его предложения по

решению рассматриваемых вопросов, обобщаются данные об их эффективности.

В случае, если разработанный проект достиг целей проектирования по всем принятым в проекте критериям, то указывается, на какие прочие факторы, ситуации, затруднения следует обратить внимание при его реализации. В случае, если разработанный проект не достиг целей проектирования по какому-либо принятому в проекте критерию, то указывается, характер этого затруднения, и, в случае его методического характера, делается вывод об обнаружении научной проблемы.

В состав курсовой работы могут быть включены приложения, содержащие нормативные и справочные материалы предприятия, выдержки из публичных нормативных актов, ГОСТов и других методических документов, с помощью которых осуществлялось проектирование. В обязательном порядке в состав курсовой работы входят список используемой литературы и содержание основных разделов работы.

В соответствии с особенностями объекта проектирования структура курсовой работы может быть изменена студентом после консультаций с руководителем.

2. ПОРЯДОК РАБОТЫ НАД КУРСОВОЙ РАБОТОЙ

В ходе написания курсовой работы должны быть выполнены следующие этапы работы:

- закрепление студента за руководителем курсовой работы;
- выбор объекта курсовой работы;
- выполнение первой части курсовой работы;
- выбор и согласование предмета моделирования программного обеспечения информационной системы;
- составление примерной структуры и плана остальных частей курсовой работы;
- выполнение курсовой работы, ее оформление и представление ее на кафедру;
- допуск курсовой работы к защите;
- подготовка курсовой работы к защите;
- защита курсовой работы.

Сроки выполнения этапов курсовой работы регламентируются графиком учебного процесса, рабочего учебного плана и устанавливаются выпускающей кафедрой.

3. РЕКОМЕНДАЦИИ ПО ВЫБОРУ ТЕМЫ КУРСОВОЙ РАБОТЫ

Рекомендуемая структура названия курсовой работы: «Разработка модели программного обеспечения «ЧЕГО»».

Объект моделирования рекомендуется сформулировать в соответствии с выбранной деятельностью предприятия.

Виды объектов могут быть определены в зависимости от:

- типа бизнес-процесса, в котором они обнаружены;
- элемента деятельности, который не соответствует норме деятельности.

По типу бизнес-процесса объекты бывают:

- в согласовании заказа;
- в проектировании;
- в обеспечении реализуемости проекта;
- во взаимодействии структур управления предприятием.

4. ОФОРМЛЕНИЕ КУРСОВОЙ РАБОТЫ

Пояснительная записка оформляется в соответствии с требованиями и правилами оформления отчетов о научно-исследовательской работе (ГОСТ 7.32-2001).

Работу оформляют на компьютере с помощью текстового редактора Word и распечатывают с одной стороны листа белой бумаги формата А4 (размером 210 x 297 мм).

Текст работы должен иметь следующие поля: левое – 25 мм, верхнее, нижнее – 20 мм, правое – 10 мм (при оформлении компонентов УМК оставляются поля в соответствии с требованиями офсетной лаборатории). Шрифт – Times New Roman, размер шрифта – 14, в т.ч. и для нумерации страниц. Используется полуторный междустрочный интервал.

Для акцентирования внимания на определенных терминах, формулах разрешается использование в работах курсива, выделение жирным шрифтом. Подчеркивания не допускаются.

Основной текст работы должен быть выровнен по ширине. Следует использовать автоматическую расстановку переносов в словах. Абзацные отступы должны быть одинаковы во всей работе и равны пяти знакам.

Сокращение слов в тексте не допускается, за исключением общепринятых. Нельзя употреблять в тексте знаки (<, >, =, №, %) без цифр, а также использовать в тексте математический знак минус (-) перед отрицательными значениями величин, нужно писать слово «минус».

В тексте должны использоваться только арабские цифры. Допускается нумерация кварталов, полугодий римскими цифрами. При записи десятичных дробей целая часть числа от дробной должна отделяться запятой (например, 15, 6 тыс. руб., 18, 5 кв. м).

Внутри текста работы не допускается использование фамилий без инициалов. Инициалы всегда (кроме Списка использованных источников) должны стоять перед фамилией через пробел. (Например, И.И. Иванов)

При ссылке в тексте на название документов органов власти с прописной буквы пишутся: Конституция Российской Федерации, Федеральный конституционный закон, Федеральный закон, Закон Российской Федерации, Указ Президента Российской Федерации, Кодекс Российской Федерации об административных правонарушениях. Со строчной буквы пишутся: проект федерального конституционного закона, проект федерального закона, распоряжение, постановление Государственной Думы Федерального Собрания, постановление Правительства Российской Федерации, распоряжение Правительства Российской Федерации. Утвержденные нормативные акты должны содержать название, дату принятия и зарегистрированный номер документа. При повторной ссылке на нормативные документы без указания их названия и номера, статус документа пишется с прописной буквы.

Пример 1

В соответствии с Федеральным законом «О федеральном бюджете на 2008 год и на плановый период 2009 и 2010 годов» № 198-ФЗ от 24 июля 2007 г.

Таким образом, вышеупомянутым федеральным законом установлены пропорции распределения средств федерального бюджета на 2008 год.

Для пояснения отдельных данных, приведенных в тексте, следует использовать сноски. Надстрочный знак сноски в виде арабских цифр со скобкой ставят непосредственно после того слова, числа, символа, предложения, к которому дается пояснение. Нумерация сносок – отдельная для каждой страницы.

Заголовки глав (разделов) и параграфов, а также таких структурных частей работы, как Содержание, Введение, Заключение, Список использованных источников, печатаются с абзацного отступа строчными буквами (кроме первой прописной). Переносы слов в заголовках не допускаются.

Подчеркивание заголовков не допускается. Точку в конце заголовка не ставят. Если заголовок состоит из двух предложений, их разделяют точкой.

Главы (разделы) работы должны иметь порядковые номера, обозначенные арабскими цифрами с точкой. Параграфы нумеруются в пределах каждой главы. Номера параграфов состоят из номера главы (раздела) и параграфа, разделенных точкой. В конце номера параграфа также должна ставиться точка, затем пробел и сам заголовок.

Пример 2

1. Теоретические и правовые основы

1.1. Понятие моделирования программного обеспечения

Расстояние между заголовком главы (раздела) и заголовком параграфа должно составлять один полуторный междустрочный интервал; между заголовком параграфа, другой структурной части работы и текстом – два полуторных междустрочных интервала.

Каждую главу (раздел) работы, а также Введение, Заключение, Список использованных источников, Приложение следует начинать с новой страницы. Параграф оформляют на новой странице только в том случае, если от текста предыдущего параграфа не осталось на листе места хотя бы для одной строки после заголовка этого параграфа.

Названия и нумерация глав (разделов), параграфов в тексте работы и в Содержании должны полностью совпадать.

Все страницы работы, за исключением приложений, нумеруются арабскими цифрами. Номера страниц проставляются в правом нижнем углу (нумерация страниц в методических разработках – в соответствии с требованиями офсетной лаборатории).

Титульный лист, Содержание включаются в общую нумерацию страниц, но при этом номер страницы на соответствующих листах не проставляется. Нумерация страниц начинается с Введения, на первой странице которого печатается цифра «3».

В тексте работы обязательно должны присутствовать ссылки на используемые источники информации. Отсутствие ссылки на цитируемый источник представляет собой нарушение авторских прав.

Ссылки на литературные источники, статистические и нормативные материалы должны отражаться в квадратных скобках в конце предложения перед точкой. Приводимые в тексте авторские цитаты выделяются кавычками.

Внутри квадратных скобок фиксируется номер источника в соответствии со Списком использованных источников, запятая, после пробела обозначается символ страницы в виде буквы «с» с точкой и номер страницы, откуда была взята цитата или цифровые данные.

При оформлении ссылок на положения нормативных правовых актов в квадратных скобках вместо номера страницы указывается номер соответствующей статьи (пункта) документа с обозначением символа «ст.» («п.»).

В тексте работы могут быть приведены перечисления, которые выделяются абзацным отступом. Перед каждой позицией перечисления ставится дефис или строчная буква со скобкой, приводимая в алфавитном порядке.

Для дальнейшей детализации перечисления используют арабские цифры, после которых ставят скобку, приводя их со смещением вправо на два знака относительно перечислений, обозначенных буквами.

Пример 3

Объекты государственной собственности применительно к данному исследованию могут быть сгруппированы следующим образом:

- а) денежные средства;
- б) природные ресурсы, в том числе:
 - 1) земельные участки,
 - 2) недра,
 - 3) водные объекты,
 - 4) леса;
- в) основные фонды;
- г) акции (доли) в хозяйственных обществах.

Цифровой материал, как правило, оформляют в виде таблиц, что обеспечивает лучшую наглядность и удобство сравнения показателей. Таблицу в зависимости от ее размера обычно помещают под текстом, в котором впервые дана на нее ссылка. Если таблица не размещается в конце страницы, то она помещается на следующей странице, а свободное место заполняется текстом, следующим за таблицей. Если размер таблицы превышает одну страницу, то таблицу следует размещать в Приложении.

Каждая таблица должна иметь заголовок, точно и кратко отражающий ее содержание. Заголовок таблицы следует помещать над таблицей по центру и печатать строчными буквами (кроме первой прописной). Переносы слов в заголовках таблиц не допускаются. В конце заголовка таблицы точка не ставится.

Если таблица заимствована из книги или статьи другого автора, на нее в конце заголовка должна быть оформлена ссылка.

В правом верхнем углу над заголовком таблицы помещают надпись «Таблица» (с заглавной буквы) с указанием номера таблицы. Расстояние между словом «Таблица» и предшествующим абзацем должно составлять два полуторных междустрочных интервала, расстояние между словом «Таблица» и заголовком, а также между заголовком и самой таблицей должно составлять один полуторный интервал.

Таблицы следует нумеровать арабскими цифрами в пределах параграфов. В этом случае номер таблицы состоит из номера параграфа и порядкового номера таблицы. В работах, не предполагающих деление глав (разделов) на параграфы, номер таблицы должен состоять из двух знаков, указывающих номер главы (раздела) и порядковый номер таблицы. Между цифрами, формирующими номер таблицы, ставится точка. После номера таблицы точку не ставят. Знак «№» перед номером таблицы не ставят.

В таблицах допускается применять меньший размер шрифта, чем в тексте, и одинарный междустрочный интервал. Не допускается выделение курсивом или полужирным шрифтом заголовков граф и строк таблиц, а также самих табличных данных. Заголовки граф и строк таблицы должны начинаться с прописной буквы, а подзаголовки граф – со строчной буквы, если они составляют одно предложение с заголовком. Подзаголовки, имеющие самостоятельное значение, начинаются с прописной буквы. В конце заголовков и подзаголовков граф и строк точки не ставят.

Графу «Номер по порядку» в таблицу включать не допускается. При необходимости порядковые номера показателей таблицы указывают в первой графе, непосредственно перед наименованием показателя.

К цифровым табличным данным должны быть указаны единицы измерения. Если данные таблицы имеют разные единицы измерения, то они указываются в соответствующих заголовках (подзаголовках) граф или строк таблицы. Если все табличные данные имеют одну и ту же единицу измерения, то данную единицу, начиная с предлога «в», приводят над таблицей справа (например, в тыс. руб., в га, в кв. м, в процентах и т.п.).

Цифровые значения в графах таблиц проставляют так, чтобы разряды чисел по всей графе были расположены один под другим. В одной графе следует соблюдать одинаковое количество десятичных знаков для всех значений величин. При отсутствии отдельных данных в таблице следует ставить прочерк. Если данные графы (строки) таблицы не требуют заполнения, то следует ставить знак «X».

Для облегчения пользования таблицей допускается проводить горизонтальные линии, разграничивающие строки таблицы. Ширина таблицы должна соответствовать ширине основного текста. При превышении ширины таблицу следует размещать в альбомном формате по тексту или в Приложении.

Если в конце страницы таблица прерывается и ее продолжение будет на следующей странице, то в первой части таблицы проводят нижнюю

горизонтальную черту, ограничивающую таблицу. При переносе таблицы ее заголовков не повторяют, а пишут в правом верхнем углу над таблицей слова «Продолжение таблицы ...» и указывают ее номер. При этом в переносимой части повторяют нумерацию граф «шапки» таблицы. Не допускается при переносе отделять заголовок таблицы от самой таблицы, оставлять на странице только «шапку» таблицы без записи хотя бы одной строки табличных данных. Итоговая строка также не должна быть отделена от таблицы.

Печать основного текста после завершения таблицы начинается через два полупетровых междустрочных интервала.

В качестве иллюстраций в работах могут быть представлены чертежи, схемы, диаграммы, рисунки и т.п. Все иллюстрации обозначают в тексте словом «рисунок».

Иллюстрации могут быть выполнены на компьютере как в черно-белом, так и в цветном варианте.

Рисунки в зависимости от их размера располагают в тексте непосредственно после того абзаца, в котором данный рисунок был впервые упомянут, или на следующей странице, а при необходимости – в приложении. Положение рисунка центрируют.

Все рисунки должны иметь наименование, которое помещают под иллюстрацией. Перед наименованием печатают слово «Рис.» (с заглавной буквы), затем пробел, после чего указывают номер рисунка. Слово «Рис.» начинают печатать с абзацного отступа.

Рисунки следует нумеровать арабскими цифрами в пределах параграфов. В этом случае номер рисунка состоит из двухзначного номера параграфа и порядкового номера рисунка. В работах, не предполагающих деление глав (разделов) на параграфы, номер рисунка должен состоять из двух знаков, указывающих номер главы (раздела) и порядковый номер рисунка. Между цифрами, формирующими номер рисунка, ставится точка. После номера рисунка также должна ставиться точка, затем пробел и наименование рисунка, которое печатают строчными буквами (кроме первой прописной). Точку в конце наименования рисунка не ставят.

Следует отметить, что нумерация рисунков проводится отдельно от нумерации таблиц, например, в параграфе 1.1 может быть представлена таблица 1.1.1 и рисунок 1.1.1.

Если иллюстрация заимствована из книги или статьи, на нее в конце наименования рисунка должна быть оформлена ссылка.

При необходимости в тексте работы могут быть использованы формулы. Формулы следует выделять из текста в отдельную строку. Между текстом и следующей за ним формулой, между формулой и следующим за ним текстом должно быть расстояние, равное двум полупетровым междустрочным интервалам.

Переносить формулы на следующую строку допускается только на знаках выполняемых математических операций, причем знак в начале следующей строки повторяют. Формулы нумеруют сквозной нумерацией в

пределах параграфов. В этом случае номер формулы состоит из двухзначного номера параграфа и порядкового номера формулы. В работах, не предполагающих деление глав (разделов) на параграфы, номер формулы должен состоять из двух знаков, указывающих номер главы (раздела) и порядковый номер формулы. Между цифрами, формирующими номер формулы, ставится точка. После номера формулы точка не ставится.

Номер печатают арабскими цифрами в круглых скобках справа от формулы на одном уровне с ней. При написании формул следует использовать буквенные символы.

Пояснения символов и числовых коэффициентов, входящих в формулу (если соответствующие пояснения не приведены ранее в тексте), приводят непосредственно под формулой. Пояснения каждого символа приводят с новой строки в той последовательности, в которой эти символы приведены в формуле. Первую строку пояснения начинают со слова «где» с двоеточием после него. После самой формулы перед пояснениями необходимо ставить запятую.

Пример 4

$$C = E \times (B + D), \quad (1.1.1)$$

где: С – размер субвенции, выделяемой субъекту Российской Федерации;

Е – численность граждан, награжденных знаком «Почетный донор СССР», «Почетный донор России», в субъекте Российской Федерации;

В – размер ежегодной денежной выплаты гражданам, награжденным знаком «Почетный донор СССР», «Почетный донор России»;

Д – расходы на оплату услуг по доставке ежегодной денежной выплаты гражданам, в пределах 1,5 процента размера ежегодной денежной выплаты.

Печать основного текста после пояснения значений символов и числовых коэффициентов формулы начинается через два полуторных междустрочных интервала.

Список использованных источников является органической частью любой работы. Он позволяет автору документально подтвердить достоверность и точность приводимых в тексте заимствований: цитат, идей, фактов, таблиц, иллюстраций, формул и других документов, на основе которых строится исследование.

Список использованных источников приводится в конце работы после Заключения, в него включаются только те источники, которые непосредственно изучались при написании работы.

Источники должны располагаться в следующем порядке:

- нормативные правовые акты;
- специальная литература;
- ресурсы Интернет.

Нормативные правовые акты должны приводиться в следующей иерархической последовательности:

- конституции (Российской Федерации, субъектов РФ);
- законы (федеральные, субъектов РФ);

- указы (Президента РФ, высших должностных лиц субъектов РФ);
- постановления (Правительства РФ, высших исполнительных органов государственной власти субъектов РФ);
- нормативные правовые акты органов местного самоуправления;
- письма, инструкции, распоряжения, приказы министерств и ведомств.

В Списке использованных источников нормативные правовые акты одинаковой юридической силы располагаются строго в хронологическом порядке по мере их принятия (от ранее принятых к более поздним принятым документам).

При библиографическом описании нормативных правовых актов сначала указывается название документа, затем его статус (например, Федеральный закон, Указ Президента РФ и т.п.), после чего приводятся дата принятия документа, его номер и дата последней редакции.

Специальная литература включает монографии, диссертации, авторефераты диссертаций, книги, учебники и учебные пособия, статистические сборники, статьи в периодических изданиях.

В Списке использованных источников специальная литература располагается строго в алфавитном порядке по фамилии авторов или, если автор не указан, по названию работы.

Информация, размещенная в сети Интернет, является электронным ресурсом удаленного доступа и может также использоваться при составлении Списка использованных источников.

Специальная литература на иностранном языке размещается в алфавитном порядке в конце Списка использованных источников. Нумерация в Списке использованных источников должна быть сплошной – от первого до последнего названия. Описание каждого источника составляется по определенной схеме и состоит из ряда обязательных элементов. В конце описания источника ставится точка.

При оформлении Списка использованных источников необходимо помнить, что перед знаками препинания никогда не ставится пробел. Пробел всегда оставляют после знаков препинания (исключение составляют только сокращения вида «М.:», «СПб.:», используемые при описании литературных источников).

Иллюстрации, таблицы, тексты вспомогательного характера могут выноситься в приложения, которые оформляются после Списка использованных источников.

Каждое приложение следует начинать с новой страницы (нумерация страниц при этом не проводится) с указанием в правом верхнем углу слова «Приложение», после которого проставляется его порядковый номер. Если приложение расположено на странице, имеющей альбомный формат, то слово «Приложение» и его порядковый номер располагают в правом верхнем углу альбомного листа.

Каждое приложение должно иметь заголовок, который записывают симметрично тексту отдельной строкой в верхней части листа. При переносе материала приложения на другую страницу в верхнем правом углу данной

страницы пишут слова «Продолжение _____ приложения...» и указывают его номер, заголовок при этом не повторяют. Нумеруются приложения в той последовательности, в которой их данные используются в работе.

Если в одно приложение входит несколько логически связанных структурных элементов, например, ряд таблиц или рисунков, то в пределах данного (т.е. одного) приложения они должны быть пронумерованы, например,

«Таблица 1», «Таблица 2» или «Рис. 1», «Рис. 2». При этом каждая таблица должна иметь свой заголовок, а рисунок – свое наименование; общий заголовок приложения в данном случае может отсутствовать.

При оформлении материалов приложений допускается использовать шрифты разной гарнитуры и размера.

5. РЕКОМЕНДАЦИИ ПО РАБОТЕ С ЛИТЕРАТУРОЙ

В работе с литературой студенту рекомендуется придерживаться следующего порядка: первоначально изучить литературу по моделированию программного обеспечения информационной системы, на ее основе определить необходимый перечень отраслей, знания которых будут востребованы в работе над настоящей курсовой работой. Затем по каждой отрасли последовательно переходить от учебной к научной литературе, далее – к справочной, и затем – к методической. В завершении рекомендуется полученные представления обобщить с использованием методологических источников.

При работе с **литературой** рекомендуется отнестись критически ко всем разработкам, имеющимся в этой области, т.к. большинство разработок – это методические разработки специалистов для неспециалистов. Такое построение методик не соответствует задачам обучения студентов высшего учебного заведения, в ходе которого студент должен овладеть способностями самостоятельно составлять такие анкеты и вопросники для конкретных управленческих ситуаций.

При работе с **учебной литературой** по отдельным отраслям, рекомендуется воспользоваться имеющимся в них представлением о структуре знаний исследуемой отрасли моделирования программного обеспечения информационной системы. Следует учесть, что учебная литература в своей основе имеет эклектичный характер изложения и служит дидактическим целям, что не позволяет использовать ее в качестве источника, обладающего концептуальной полнотой охвата имеющихся в этой области знаний и прикладным характером изложения.

При работе с **научной литературой** по представленным в методиках моделирования программного обеспечения рекомендуется проследить историческое развитие взглядов отдельных научных школ на предмет научного исследования в связи с разрешением соответствующих проблем, возникающих в практике. Такое исследование рекомендуется в целях формирования у студента собственного понимания применения имеющихся концептуальных разработок в практической деятельности предприятия и исключения неполного охвата концептуальных представлений при выборе тех или иных методик в работе над настоящей курсовой работой. Следует учесть, что научная литература характеризуется историческими условиями ее возникновения и служит целям концептуального обобщения и научной полемики, что не позволяет использовать ее в качестве источника по практическому применению предложенных в ней методов управления и общепринятому применению используемых в ней профессиональных понятий.

При работе со **справочной литературой** рекомендуется выработать правильное (общепринятое) применение понятий профессионального языка,

используемого в методах моделирования программного обеспечения информационной системы. Не следует использовать справочную литературу для исследования концептуальных обобщений, т.к. концептуальные разработки представлены в ней фрагментарно.

При работе с **методической литературой** по исследуемым отраслям моделирования программного обеспечения информационной системы, рекомендуется исследовать ее с точки зрения применения в данной конкретной управленческой ситуации предприятия, взятой за предмет курсового проектирования. Следует учесть, что методическая литература характеризуется определенным, довольно узким набором ситуаций ее применения, что не позволяет использовать ее в качестве источника по критериально обеспеченному выбору методов исследования и проектирования. Из числа методических источников особо нужно выделить **нормативные источники**, область применения которых строго определена органом, их принявшим. При этом необходимо учитывать, что нормативные акты в силу своей нормативности не позволяют создать методические конкурентные преимущества разрабатываемого проекта.

При работе с **методологической литературой** рекомендуется исследовать ее в целях обнаружения инструментов неслучайного выбора методов моделирования программного обеспечения информационной системы.

6. ТЕМАТИКА КУРСОВЫХ РАБОТ

1. Моделирование программного обеспечения информационной системы цифровой диктофон
2. Моделирование программного обеспечения информационной системы торговый автомат
3. Моделирование программного обеспечения информационной системы табло на станции метро
4. Моделирование программного обеспечения информационной системы пункта проката
5. Моделирование программного обеспечения информационной системы мини-АТС
6. Моделирование программного обеспечения информационной системы телефон
7. Моделирование программного обеспечения информационной системы стиральная машина
8. Моделирование программного обеспечения информационной системы таксофон
9. Моделирование программного обеспечения информационной системы банкомат
10. Моделирование программного обеспечения информационной системы холодильник
11. Моделирование программного обеспечения информационной системы кодовый замок
12. Моделирование программного обеспечения информационной системы турникет метро
13. Моделирование программного обеспечения информационной системы учета товаров
14. Моделирование программного обеспечения информационной системы библиотечного пункта
15. Моделирование программного обеспечения информационной системы интернет-магазин
16. Моделирование программного обеспечения информационной системы WWW-конференция
17. Моделирование программного обеспечения информационной системы каталог ресурсов Интернет
18. Моделирование программного обеспечения информационной системы будильник
19. Моделирование программного обеспечения информационной системы генеалогическое дерево
20. Моделирование программного обеспечения информационной системы телевизор
21. Моделирование программного обеспечения информационной системы поддержки составления расписания

22. Моделирование программного обеспечения информационной системы домофон
23. Моделирование программного обеспечения информационной системы оператора контактного центра
24. Моделирование программного обеспечения информационной системы инфомат
25. Моделирование программного обеспечения информационной системы бензоколонка
26. Моделирование программного обеспечения информационной системы кассовый аппарат самообслуживания
27. Моделирование программного обеспечения информационной системы автомат продажи разливных напитков
28. Моделирование программного обеспечения информационной системы автомобильный бортовой компьютер
29. Моделирование программного обеспечения информационной системы автомат продажи железнодорожных билетов
30. Моделирование программного обеспечения информационной системы аппарат мойки самообслуживания

7. ИСХОДНЫЙ МАТЕРИАЛ ДЛЯ ВЫПОЛНЕНИЯ КУРСОВОЙ РАБОТЫ

Основными источниками информация для исследования являются локальные нормативные акты предприятия (устав, приказы, инструкции, положения и т.п.) и данные специальных самостоятельно проведенных наблюдений и обследований, а также первичная документация (наряды, личные карточки и др.), планы и проекты, и пр.

При описании системы деятельности рекомендуется особое внимание уделить определению деятельности и функционала исследуемого объекта. Заказчиком могут быть следующие лица:

- потребители продукции предприятия;
- инвесторы или собственники предприятия;
- поставщики ресурсов (кредиторы, профсоюзы, менеджмент).

Перед тем как разрабатывать модель программного обеспечения, необходимо оценить спрос на товар (работы или услуги), который предлагается производить. Необходимо понять, что и кому будет продаваться и почему люди будут это покупать. Надо помнить, что люди покупают не просто продукт или услугу, они покупают совокупность определенных преимуществ, разрешая свои проблемы. Эта совокупность преимуществ включает в себя четыре элемента (специалисты по маркетингу называют их маркетинговым комплексом): характеристики самого продукта, его цену, поощрение спроса на данный продукт и место его реализации.

Если предприятие приведет свою продукцию или услуги в соответствие с этими требованиями и сделает это лучше, чем конкуренты, то оно сможет увеличить долю своего участия на рынке и, следовательно, увеличить прибыльность. Важным обстоятельством при этом будет определение покупателей: каковы их запросы и как на них выйти, довести до них свою продукцию. Без хорошего понимания запросов покупателей (клиентов) нельзя оценить сильные и слабые стороны продукции или услуг.

Необходимые данные можно получить, обратившись в соответствующие организации, можно провести собственные исследования. Источником информации могут стать публикации отраслевых ассоциаций, статьи в научных журналах, официальная отчетность аналогичных компаний на официальных сайтах.

8. ОСОБЫЕ РЕКОМЕНДАЦИИ К ВЫПОЛНЕНИЮ ОСНОВНОЙ ЧАСТИ КУРСОВОЙ РАБОТЫ

8.1 ОБЩИЕ ТРЕБОВАНИЯ

Полностью раскрыть все варианты использования. Вариантов использования должно быть не менее 15-20. Актёров (интерфейсов) не менее 3-х. Логика функционирования программного средства должна быть полностью описана. В результате должен быть сформирован список актёров (интерфейсов) и их функций (вариантов использования) в виде уровневого списка или таблицы. А так же должны быть сформированы диаграмма вариантов использования и диаграмма кооперации, полностью описывающие функциональное назначение программного средства.

8.2 ОПИСАНИЕ ДИАГРАММЫ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ

Визуальное моделирование в UML можно представить как некоторый процесс поуровневого спуска от наиболее общей и абстрактной концептуальной модели исходной системы к логической, а затем и к физической модели соответствующей программной системы. Для достижения этих целей вначале строится модель в форме так называемой диаграммы вариантов использования (use case diagram), которая описывает функциональное назначение системы или, другими словами, то, что система будет делать в процессе своего функционирования. Диаграмма вариантов использования является исходным концептуальным представлением или концептуальной моделью системы в процессе ее проектирования и разработки.

Разработка диаграммы вариантов использования преследует цели:

- Определить общие границы и контекст моделируемой предметной области на начальных этапах проектирования системы.
- Сформулировать общие требования к функциональному поведению проектируемой системы.
- Разработать исходную концептуальную модель системы для ее последующей детализации в форме логических и физических моделей.
- Подготовить исходную документацию для взаимодействия разработчиков системы с ее заказчиками и пользователями.

Суть данной диаграммы состоит в следующем: проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью так называемых вариантов использования. При этом актером (actor) или действующим лицом называется любая сущность, взаимодействующая с системой извне. Это может быть человек, техническое устройство, программа или любая другая система, которая может служить источником воздействия на моделируемую систему так, как определит сам разработчик. В свою очередь, вариант использования (use case) служит для описания сервисов, которые система предоставляет актеру. Другими словами, каждый вариант использования определяет некоторый набор действий, совершаемый системой при диалоге с актером. При этом ничего не говорится о том, каким образом будет реализовано взаимодействие актеров с системой.

В самом общем случае, диаграмма вариантов использования представляет собой граф специального вида, который является графической нотацией для представления конкретных вариантов использования, актеров, возможно некоторых интерфейсов, и отношений между этими элементами. При этом отдельные компоненты диаграммы могут быть заключены в прямоугольник, который обозначает проектируемую систему в целом. Следует отметить, что отношениями данного графа могут быть только некоторые фиксированные типы взаимосвязей между актерами и вариантами использования, которые в совокупности описывают сервисы или функциональные требования к моделируемой системе.

Как было отмечено ранее, рациональный унифицированный процесс разработки модели сложной системы представляет собой разбиение ее на составные части с минимумом взаимных связей на основе выделения пакетов. В самом языке UML пакет Варианты использования является подпакетом пакета Элементы поведения. Последний специфицирует понятия, при помощи которых определяют функциональность моделируемых систем. Элементы пакета вариантов использования являются первичными по отношению к тем, с помощью которых могут быть описаны сущности, такие как системы и подсистемы. Однако внутренняя структура этих сущностей никак не описывается. Базовые элементы этого пакета – вариант использования и актер.

8.3 ВАРИАНТ ИСПОЛЬЗОВАНИЯ

Конструкция или стандартный элемент языка UML вариант использования применяется для спецификации общих особенностей поведения системы или любой другой сущности предметной области без рассмотрения внутренней структуры этой сущности. Каждый вариант использования определяет последовательность действий, которые должны быть выполнены проектируемой системой при взаимодействии ее с соответствующим актером. Диаграмма вариантов может дополняться пояснительным текстом, который раскрывает смысл или семантику составляющих ее компонентов. Такой пояснительный текст получил название примечания или сценария.

Отдельный вариант использования обозначается на диаграмме эллипсом, внутри которого содержится его краткое название или имя в форме глагола с пояснительными словами.

Цель варианта использования заключается в том, чтобы определить законченный аспект или фрагмент поведения некоторой сущности без раскрытия внутренней структуры этой сущности. В качестве такой сущности может выступать исходная система или любой другой элемент модели, который обладает собственным поведением, подобно подсистеме или классу в модели системы.

Каждый вариант использования соответствует отдельному сервису, который предоставляет моделируемую сущность или систему по запросу пользователя (актера), т. е. определяет способ применения этой сущности. Сервис, который инициализируется по запросу пользователя, представляет собой законченную последовательность действий. Это означает, что после того как система закончит обработку запроса пользователя, она должна возвратиться в исходное состояние, в котором готова к выполнению следующих запросов.

Варианты использования описывают не только взаимодействия между пользователями и сущностью, но также реакции сущности на получение отдельных сообщений от пользователей и восприятие этих сообщений за пределами сущности. Варианты использования могут включать в себя описание особенностей способов реализации сервиса и различных исключительных ситуаций, таких как корректная обработка ошибок системы. Множество вариантов использования в целом должно определять все возможные стороны ожидаемого поведения системы. Для удобства множество вариантов использования может рассматриваться как отдельный пакет.

С системно-аналитической точки зрения варианты использования могут применяться как для спецификации внешних требований к проектируемой системе, так и для спецификации функционального поведения уже существующей системы. Кроме этого, варианты использования неявно устанавливают требования, определяющие, как пользователи должны

взаимодействовать с системой, чтобы иметь возможность корректно работать с предоставляемыми данной системой сервисами!

Применение вариантов использования на всех уровнях диаграммы позволяет не только достичь требуемого уровня унификации обозначений для представления функциональности подсистем и системы в целом, но и является мощным средством последовательного уточнения требований к проектируемой системе на основе полууровневого спуска от пакетов системы к операциям классов. С другой стороны, модификация отдельных операций класса может оказать обратное влияние на уточнение сервиса соответствующего варианта использования, т. е. реализовать эффект обратной связи с целью уточнения спецификаций или требований на уровне пакетов системы.

В метамодели UML вариант использования является подклассом классификатора, который описывает последовательности действий, выполняемых отдельным экземпляром варианта использования. Эти действия включают изменения состояния и взаимодействия со средой варианта использования. Эти последовательности могут описываться различными способами, включая такие, как графы деятельности и автоматы.

Примерами вариантов использования могут являться следующие действия: проверка состояния текущего счета клиента, оформление заказа на покупку товара, получение дополнительной информации о кредитоспособности клиента, отображение графической формы на экране монитора и другие действия.

8.4 АКТЕРЫ

Актер представляет собой любую внешнюю по отношению к моделируемой системе сущность, которая взаимодействует с системой и использует ее функциональные возможности для достижения определенных целей или решения частных задач. При этом актеры служат для обозначения согласованного множества ролей, которые могут играть пользователи в процессе взаимодействия с проектируемой системой. Каждый актер может рассматриваться как некая отдельная роль относительно конкретного варианта использования. Стандартным графическим обозначением актера на диаграммах является фигурка «человечка», под которой записывается конкретное имя актера.

В некоторых случаях актер может обозначаться в виде прямоугольника класса с ключевым словом «актер» и обычными составляющими элементами класса. Имена актеров должны записываться заглавными буквами и следовать рекомендациям использования имен для типов и классов модели. При этом символ отдельного актера связывает соответствующее описание актера с конкретным именем. Имена абстрактных актеров, как и других абстрактных элементов языка UML, рекомендуется обозначать курсивом.

Примерами актеров могут быть: клиент банка, банковский служащий, продавец магазина, менеджер отдела продаж, пассажир авиарейса, водитель автомобиля, администратор гостиницы, сотовый телефон и другие сущности, имеющие отношение к концептуальной модели соответствующей предметной области.

Актеры используются для моделирования внешних по отношению к проектируемой системе сущностей, которые взаимодействуют с системой и используют ее в качестве отдельных пользователей. В качестве актеров могут выступать другие системы, подсистемы проектируемой системы или отдельные классы. Важно понимать, что каждый актер определяет некоторое согласованное множество ролей, в которых могут выступать пользователи данной системы в процессе взаимодействия с ней. В каждый момент времени с системой взаимодействует вполне определенный пользователь, при этом он играет или выступает в одной из таких ролей. Наиболее наглядный пример актера – конкретный пользователь системы со своими собственными параметрами аутентификации.

Любая сущность, которая согласуется с подобным неформальным определением актера, представляет собой экземпляр или пример актера. Для моделируемой системы актерами могут быть как субъекты-пользователи, так и другие системы. Поскольку пользователи системы всегда являются внешними по отношению к этой системе, то они всегда представляются в виде актеров.

Так как в общем случае актер всегда находится вне системы, его внутренняя структура никак не определяется. Для актера имеет значение только его внешнее представление, т. е. то, как он воспринимается со стороны системы. Актеры взаимодействуют с системой посредством

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

Два и более актера могут иметь общие свойства, т. е. взаимодействовать с одним и тем же множеством вариантов использования одинаковым образом. Такая общность свойств и поведения представляется в виде рассматриваемого ниже отношения обобщения с другим, возможно, абстрактным актером, который моделирует соответствующую общность ролей.

8.5 ИНТЕРФЕЙСЫ

Интерфейс (interface) служит для спецификации параметров модели, которые видимы извне без указания их внутренней структуры. В языке UML интерфейс является классификатором и характеризует только ограниченную часть поведения моделируемой сущности. Применительно к диаграммам вариантов использования, интерфейсы определяют совокупность операций, которые обеспечивают необходимый набор сервисов или функциональности для актеров. Интерфейсы не могут содержать ни атрибутов, ни состояний, ни направленных ассоциаций. Они содержат только операции без указания особенностей их реализации. Формально интерфейс эквивалентен абстрактному классу без атрибутов и методов с наличием только абстрактных операций.

На диаграмме вариантов использования интерфейс изображается в виде маленького круга, рядом с которым записывается его имя. В качестве имени может быть существительное, которое характеризует соответствующую информацию или сервис (например, «датчик», «сирена», «видеокамера»), но чаще строка текста (например, «запрос к базе данных», «форма ввода», «устройство подачи звукового сигнала»). Если имя записывается на английском, то оно должно начинаться с заглавной буквы I.

Графический символ отдельного интерфейса может соединяться на диаграмме сплошной линией с тем вариантом использования, который его поддерживает. Сплошная линия в этом случае указывает на тот факт, что связанный с интерфейсом вариант использования должен реализовывать все операции, необходимые для данного интерфейса, а возможно и больше. Кроме этого, интерфейсы могут соединяться с вариантами использования пунктирной линией со стрелкой, означающей, что вариант использования предназначен для спецификации только того сервиса, который необходим для реализации данного интерфейса.

С системно-аналитической точки зрения интерфейс не только отделяет спецификацию операций системы от их реализации, но и определяет общие границы проектируемой системы. В последующем интерфейс может быть уточнен явным указанием тех операций, которые специфицируют отдельный аспект поведения системы. В этом случае он изображается в форме прямоугольника класса с ключевым словом «interface» в секции имени, с пустой секцией атрибутов и с непустой секцией операций. Однако подобное графическое представление используется на диаграммах классов или диаграммах, характеризующих поведение моделируемой системы.

Важность интерфейсов заключается в том, что они определяют стыковочные узлы в проектируемой системе, что совершенно необходимо для организации коллективной работы над проектом. Более того, спецификация интерфейсов способствует «безболезненной» модификации уже существующей системы при переходе на новые технологические решения. В этом случае изменению подвергается только реализация операций, но никак не функциональность самой системы.

8.6 ПРИМЕЧАНИЯ

Примечания (notes) в языке UML предназначены для включения в модель произвольной текстовой информации, имеющей непосредственное отношение к контексту разрабатываемого проекта. В качестве такой информации могут быть комментарии разработчика (например, дата и версия разработки диаграммы или ее отдельных компонентов), ограничения (например, на значения отдельных связей или экземпляры сущностей) и помеченные значения. Применительно к диаграммам вариантов использования примечание может носить самую общую информацию, относящуюся к общему контексту системы.

Графически примечания обозначаются прямоугольником с «загнутым» верхним правым уголком. Внутри прямоугольника содержится текст примечания. Примечание может относиться к любому элементу диаграммы, в этом случае их соединяет пунктирная линия. Если примечание относится к нескольким элементам, то от него проводятся, соответственно, несколько линий. Разумеется, примечания могут присутствовать не только на диаграмме вариантов использования, но и на других канонических диаграммах.

Если в примечании указывается ключевое слово «constraint», то данное примечание является ограничением, налагаемым на соответствующий элемент модели, но не на саму диаграмму. При этом запись ограничения заключается в фигурные скобки и должна соответствовать правилам правильного построения выражений языка OCL. Более подробно язык объектных ограничений и примеры его использования будут рассмотрены в приложении. Однако для диаграмм вариантов использования ограничения включать в модели не рекомендуется, поскольку они достаточно жестко регламентируют отдельные аспекты системы. Подобная регламентация противоречит неформальному характеру общей модели системы, в качестве которой выступает диаграмма вариантов использования.

8.7 ОТНОШЕНИЯ

Между компонентами диаграммы вариантов использования могут существовать различные отношения, которые описывают взаимодействие экземпляров одних актеров и вариантов использования с экземплярами других актеров и вариантов. Один актер может взаимодействовать с несколькими вариантами использования. В этом случае этот актер обращается к нескольким сервисам данной системы. В свою очередь один вариант использования может взаимодействовать с несколькими актерами, предоставляя для всех них свой сервис. Следует заметить, что два варианта использования, определенные для одной и той же сущности, не могут взаимодействовать друг с другом, поскольку каждый из них самостоятельно описывает законченный вариант использования этой сущности. Более того, варианты использования всегда предусматривают некоторые сигналы или сообщения, когда взаимодействуют с актерами за пределами системы. В то же время могут быть определены другие способы для взаимодействия с элементами внутри системы.

В языке UML имеется несколько стандартных видов отношений между актерами и вариантами использования:

- Отношение ассоциации (association relationship)
- Отношение расширения (extend relationship)
- Отношение обобщения (generalization relationship)
- Отношение включения (include relationship)

При этом общие свойства вариантов использования могут быть представлены тремя различными способами, а именно с помощью отношений расширения, обобщения и включения.

Отношение ассоциации является одним из фундаментальных понятий в языке UML и в той или иной степени используется при построении всех графических моделей систем в форме канонических диаграмм.

Применительно к диаграммам вариантов использования оно служит для обозначения специфической роли актера в отдельном варианте использования. Другими словами, ассоциация специфицирует семантические особенности взаимодействия актеров и вариантов использования в графической модели системы. Таким образом, это отношение устанавливает, какую конкретную роль играет актер при взаимодействии с экземпляром варианта использования. На диаграмме вариантов использования, так же как и на других диаграммах, отношение ассоциации обозначается сплошной линией между актером и вариантом использования. Эта линия может иметь дополнительные условные обозначения, такие, например, как имя и кратность.

Кратность (multiplicity) ассоциации указывается рядом с обозначением компонента диаграммы, который является участником данной ассоциации. Кратность характеризует общее количество конкретных экземпляров данного компонента, которые могут выступать в качестве элементов данной ассоциации. Применительно к диаграммам вариантов использования

кратность имеет специальное обозначение в форме одной или нескольких цифр и, возможно, специального символа "*" (звездочка).

Отношение расширения определяет взаимосвязь экземпляров отдельного варианта использования с более общим вариантом, свойства которого определяются на основе способа совместного объединения данных экземпляров. В метамодели отношение расширения является направленным и указывает, что применительно к отдельным примерам некоторого варианта использования должны быть выполнены конкретные условия, определенные для расширения данного варианта использования. Так, если имеет место отношение расширения от варианта использования А к варианту использования В, то это означает, что свойства экземпляра варианта использования В могут быть дополнены благодаря наличию свойств у расширенного варианта использования А.

Отношение расширения между вариантами использования 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» («расширяет»).

Отношение расширения отмечает тот факт, что один из вариантов использования может присоединять к своему поведению некоторое дополнительное поведение, определенное для другого варианта использования. Данное отношение включает в себя некоторое условие и ссылки на точки расширения в базовом варианте использования. Чтобы расширение имело место, должно быть выполнено определенное условие данного отношения. Ссылки на точки расширения определяют те места в базовом варианте использования, в которые должно быть помещено соответствующее расширение при выполнении условия.

Отношение обобщения служит для указания того факта, что некоторый вариант использования А может быть обобщен до варианта использования В. В этом случае вариант А будет являться специализацией варианта В. При этом В называется предком или родителем по отношению А, а вариант А – потомком по отношению к варианту использования В. Следует подчеркнуть, что потомок наследует все свойства и поведение своего родителя, а также может быть дополнен новыми свойствами и особенностями поведения. Графически данное отношение обозначается сплошной линией со стрелкой в форме незакрашенного треугольника, которая указывает на родительский вариант использования. Эта линия со стрелкой имеет специальное название – стрелка «обобщение».

Отношение обобщения между вариантами использования применяется в том случае, когда необходимо отметить, что дочерние варианты использования обладают всеми атрибутами и особенностями поведения родительских вариантов. При этом дочерние варианты использования участвуют во всех отношениях родительских вариантов. В свою очередь, дочерние варианты могут наделяться новыми свойствами поведения,

которые отсутствуют у родительских вариантов использования, а также уточнять или модифицировать наследуемые от них свойства поведения.

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. Данное отношение является направленным бинарным отношением в том смысле, что пара экземпляров вариантов использования всегда упорядочена в отношении включения.

Семантика этого отношения определяется следующим образом. Когда экземпляр первого варианта использования в процессе своего выполнения достигает точки включения в последовательность поведения экземпляра второго варианта использования, экземпляр первого варианта использования выполняет последовательность действий, определяющую поведение экземпляра второго варианта использования, после чего продолжает выполнение действий своего поведения. При этом предполагается, что даже если экземпляр первого варианта использования может иметь несколько включаемых в себя экземпляров других вариантов, выполняемые ими действия должны закончиться к некоторому моменту, после чего должно быть продолжено выполнение прерванных действий экземпляра первого варианта использования в соответствии с заданным для него поведением.

Один вариант использования может быть включен в несколько других вариантов, а также включать в себя другие варианты. Включаемый вариант использования может быть независимым от базового варианта в том смысле, что он предоставляет последнему некоторое инкапсулированное поведение, детали реализации которого скрыты от последнего и могут быть легко перераспределены между несколькими включаемыми вариантами использования. Более того, базовый вариант может зависеть только от результатов выполнения включаемого в него поведения, но не от структуры включаемых в него вариантов.

Отношение включения, направленное от варианта использования А к варианту использования В, указывает, что каждый экземпляр варианта А включает в себя функциональные свойства, заданные для варианта В. Эти свойства специализируют поведение соответствующего варианта А на данной диаграмме. Графически данное отношение обозначается пунктирной линией со стрелкой (вариант отношения зависимости), направленной от базового варианта использования к включаемому. При этом данная линия со стрелкой помечается ключевым словом «include» («включает»).

8.8 ДИАГРАММА КООПЕРАЦИИ

Особенности взаимодействия элементов моделируемой системы могут быть представлены на диаграммах последовательности и кооперации. Если первая служит для визуализации временных аспектов взаимодействия, то диаграмма кооперации предназначена для спецификации структурных аспектов взаимодействия. Главная особенность диаграммы кооперации заключается в возможности графически представить не только последовательность взаимодействия, но и все структурные отношения между объектами, участвующими в этом взаимодействии.

Прежде всего, на диаграмме кооперации в виде прямоугольников изображаются участвующие во взаимодействии объекты, содержащие имя объекта, его класс и, возможно, значения атрибутов. Далее, как и на диаграмме классов, указываются ассоциации между объектами в виде различных соединительных линий. При этом можно явно указать имена ассоциации и ролей, которые играют объекты в данной ассоциации. Дополнительно могут быть изображены динамические связи – потоки сообщений. Они представляются также в виде соединительных линий между объектами, над которыми располагается стрелка с указанием направления, имени сообщения и порядкового номера в общей последовательности инициализации сообщений.

В отличие от диаграммы последовательности, на диаграмме кооперации изображаются только отношения между объектами, играющими определенные роли во взаимодействии. С другой стороны, на этой диаграмме не указывается время в виде отдельного измерения. Поэтому последовательность взаимодействий и параллельных потоков может быть определена с помощью порядковых номеров. Следовательно, если необходимо явно специфицировать взаимосвязи между объектами в реальном времени, лучше это делать на диаграмме последовательности.

Поведение системы может описываться на уровне отдельных объектов, которые обмениваются между собой сообщениями, чтобы достичь нужной цели или реализовать некоторый сервис. С точки зрения аналитика или конструктора важно представить в проекте системы структурные связи отдельных объектов между собой. Такое статическое представление структуры системы как совокупности взаимодействующих объектов и обеспечивает диаграмма кооперации.

Таким образом, с помощью диаграммы кооперации можно описать полный контекст взаимодействий как своеобразный временной «среза» совокупности объектов, взаимодействующих между собой для выполнения определенной задачи или бизнес-цели программной системы.

Понятие кооперации (collaboration) является одним из фундаментальных понятий в языке UML. Оно служит для обозначения множества взаимодействующих с определенной целью объектов в общем контексте моделируемой системы. Цель самой кооперации состоит в том, чтобы специфицировать особенности реализации отдельных наиболее значимых

операций в системе. Кооперация определяет структуру поведения системы в терминах взаимодействия участников этой кооперации.

Кооперация может быть представлена на двух уровнях:

- На уровне спецификации – показывает роли классификаторов и роли ассоциаций в рассматриваемом взаимодействии.

- На уровне примеров – указывает экземпляры и связи, образующие отдельные роли в кооперации.

Диаграмма кооперации уровня спецификации показывает роли, которые играют участвующие во взаимодействии элементы. Элементами кооперации на этом уровне являются классы и ассоциации, которые обозначают отдельные роли классификаторов и ассоциации между участниками кооперации.

Диаграмма кооперации уровня примеров представляется совокупностью объектов (экземпляры классов) и связей (экземпляры ассоциаций). При этом связи дополняются стрелками сообщений. На данном уровне показываются только релевантные объекты, т. е. имеющие непосредственное отношение к реализации операции или классификатора.

В кооперации уровня примеров определяются свойства, которые должны иметь экземпляры для того, чтобы участвовать в кооперации. Кроме свойств объектов на диаграмме кооперации также указываются ассоциации, которые должны иметь место между объектами кооперации. При этом вовсе не обязательно изображать все свойства или все ассоциации, поскольку на диаграмме кооперации присутствуют только роли классификаторов, но не сами классификаторы. Таким образом, в то время как классификатор требует полного описания всех своих экземпляров, роль классификатора требует описания только тех свойств и ассоциаций, которые необходимы для участия в отдельной кооперации.

Отсюда вытекает важное следствие. Одна и та же совокупность объектов может участвовать в различных кооперациях. При этом, в зависимости от рассматриваемой кооперации, могут изменяться как свойства отдельных объектов, так и связи между ними. Именно это отличает диаграмму кооперации от диаграммы классов, на которой должны быть указаны все свойства и ассоциации между элементами диаграммы.

8.9 ЗАКЛЮЧИТЕЛЬНЫЕ РЕКОМЕНДАЦИИ ПО РАЗРАБОТКЕ ДИАГРАММ ВАРИАНТОВ ИСПОЛЬЗОВАНИЯ

Главное назначение диаграммы вариантов использования заключается в формализации функциональных требований к системе с помощью понятий соответствующего пакета и возможности согласования полученной модели с заказчиком на ранней стадии проектирования. Любой из вариантов использования может быть подвергнут дальнейшей декомпозиции на множество подвариантов использования отдельных элементов, которые образуют исходную сущность. Рекомендуемое общее количество актеров в модели – не более 20, а вариантов использования – не более 50. В противном случае модель теряет свою наглядность и, возможно, заменяет собой одну из некоторых других диаграмм.

Семантика построения диаграммы вариантов использования должна определяться следующими особенностями рассмотренных выше элементов модели. Отдельный экземпляр варианта использования по своему содержанию является выполнением последовательности действий, которая инициализируется посредством экземпляра сообщения от экземпляра актера. В качестве отклика или ответной реакции на сообщение актера экземпляр варианта использования выполняет последовательность действий, установленную для данного варианта использования. Экземпляры актеров могут генерировать новые экземпляры сообщений для экземпляров вариантов использования.

Подобное взаимодействие будет продолжаться до тех пор, пока не закончится выполнение требуемой последовательности действий экземпляром варианта использования, и соответствующий экземпляр актера (и никакой другой) не получит требуемый экземпляр сервиса. Окончание взаимодействия означает отсутствие инициализации экземпляров сообщений от экземпляров актеров для соответствующих экземпляров вариантов использования.

Варианты использования могут быть специфицированы в виде текста, а в последующем – с помощью операций и методов вместе с атрибутами, в виде графа деятельности, посредством автомата или любого другого механизма описания поведения, включающего предусловия и постусловия. Взаимодействие между вариантами использования и актерами может уточняться на диаграмме кооперации, когда описываются взаимосвязи между сущностью, содержащей эти варианты использования, и окружением или внешней средой этой сущности.

В случае, когда для представления иерархической структуры проектируемой системы используются подсистемы, система может быть определена в виде вариантов использования на всех уровнях. Отдельные подсистемы или классы могут выступать в роли таких вариантов использования. При этом вариант, соответствующий некоторому из этих элементов, в последующем может уточняться множеством более мелких вариантов использования, каждый из которых будет определять сервис

элемента модели, содержащийся в сервисе исходной системы. Вариант использования в целом может рассматриваться как суперсервис для уточняющих его подвариантов, которые, в свою очередь, могут рассматриваться как подсервисы исходного варианта использования.

Функциональность, определенная для более общего варианта использования, полностью наследуется всеми вариантами нижних уровней. Однако следует заметить, что структура элемента-контейнера не может быть представлена вариантами использования, поскольку они могут представлять только функциональность отдельных элементов модели. Подчиненные варианты использования кооперируются для совместного выполнения суперсервиса варианта использования верхнего уровня. Эта кооперация также может быть представлена на диаграмме кооперации в виде совместных действий отдельных элементов модели.

Отдельные варианты использования нижнего уровня могут участвовать в нескольких кооперациях, т. е. играть определенную роль при выполнении сервисов нескольких вариантов верхнего уровня. Для отдельных таких коопераций могут быть определены соответствующие роли актеров, взаимодействующих с конкретными вариантами использования нижнего уровня. Эти роли будут играть актеры нижнего уровня модели системы. Хотя некоторые из таких актеров могут быть актерами верхнего уровня, это не противоречит принятым в языке UML семантическим правилам построения диаграмм вариантов использования. Более того, интерфейсы вариантов использования верхнего уровня могут полностью совпадать по своей структуре с соответствующими интерфейсами вариантов нижнего уровня.

Окружение вариантов использования нижнего уровня является самостоятельным элементом модели, который в свою очередь содержит другие элементы модели, определенные для этих вариантов использования. Таким образом, с точки зрения общего представления верхнего уровня взаимодействие между вариантами использования нижнего уровня определяет результат выполнения сервиса варианта верхнего уровня. Отсюда следует, что вариант использования является элементом-контейнером.

Варианты использования классов соответствуют операциям этого класса, поскольку сервис класса является по существу выполнением операций данного класса. Некоторые варианты использования могут соответствовать применению только одной операции, в то время как другие – конечного множества операций, определенных в виде последовательности операций. В то же время одна операция может быть необходима для выполнения нескольких сервисов класса и поэтому будет появляться в нескольких вариантах использования этого класса.

Реализация варианта использования зависит от типа элемента модели, в котором он определен. Например, поскольку варианты использования класса определяются посредством операций этого класса, они реализуются соответствующими методами. С другой стороны, варианты использования подсистемы реализуются элементами, из которых состоит данная подсистема. Поскольку подсистема не имеет своего собственного поведения,

все предлагаемые подсистемой сервисы должны представлять собой композицию сервисов, предлагаемых отдельными элементами этой подсистемы, т.е., в конечном итоге, классами. Эти элементы могут взаимодействовать друг с другом для совместного обеспечения требуемого поведения отдельного варианта использования. Такое совместное обеспечение требуемого поведения описывается специальным элементом языка UML – кооперация или сотрудничество на диаграммах кооперации. Отметим, что кооперации используются как для уточнения спецификаций в виде вариантов использования нижних уровней диаграммы, так и для описания особенностей их последующей реализации.

Если в качестве моделируемой сущности выступает система или подсистема самого верхнего уровня, то отдельные пользователи вариантов использования этой системы моделируются актерами. Такие актеры, являясь внутренними по отношению к моделируемым подсистемам нижних уровней, часто в явном виде не указываются, хотя и присутствуют неявно в модели подсистемы. Вместо этого варианты использования непосредственно обращаются к тем модельным элементам, которые содержат в себе подобные неявные актеры, т.е. экземпляры которых играют роли таких актеров при взаимодействии с вариантами использования. Эти модельные элементы могут содержаться в других пакетах или подсистемах. В последнем случае роли определяются в том пакете, к которому относится соответствующая подсистема.

С системно-аналитической точки зрения построение диаграммы вариантов использования специфицирует не только функциональные требования к проектируемой системе, но и выполняет исходную структуризацию предметной области. Последняя задача сочетает в себе не только следование техническим рекомендациям, но и является в некотором роде искусством, умением выделять главное в модели системы. Хотя рациональный унифицированный процесс не исключает итеративный возврат в последующем к диаграмме вариантов использования для ее модификации, не вызывает сомнений тот факт, что любая подобная модификация потребует, как по цепочке, изменений во всех других представлениях системы. Поэтому всегда необходимо стремиться к возможно более точному представлению модели в форме диаграммы вариантов использования.

Если же варианты использования применяются для спецификации части системы, то они будут эквивалентны соответствующим вариантам использования в модели подсистемы для части соответствующего пакета. Важно понимать, что все сервисы системы должны быть явно определены на диаграмме вариантов использования, и никаких других сервисов, которые отсутствуют на данной диаграмме, проектируемая система не может выполнять по определению. Более того, если для моделирования реализации системы используются сразу несколько моделей (например, модель анализа и модель проектирования), то множество вариантов использования всех пакетов системы должно быть эквивалентно множеству вариантов использования модели в целом.

8.10 ЗАКЛЮЧИТЕЛЬНЫЕ РЕКОМЕНДАЦИИ ПО ПОСТРОЕНИЮ ДИАГРАММ КООПЕРАЦИИ

Построение диаграммы кооперации можно начинать сразу после построения диаграммы вариантов использования. В этом случае каждый из вариантов использования может быть специфицирован в виде отдельной диаграммы кооперации уровня спецификации. Эта диаграмма способствует более полному пониманию особенностей реализации функций системой, хотя и не может содержать всю информацию, необходимую для их реализации.

В дальнейшем, после построения диаграммы классов, каждая из диаграмм кооперации может уточняться в виде соответствующей диаграммы уровня примеров. Важно понимать, что диаграмма кооперации этого уровня может содержать те и только те объекты и связи, которые уже определены на построенной ранее диаграмме классов. В противном случае, если возникает необходимость включения в диаграмму кооперации объектов, которые создаются на основе отсутствующих классов, соответствующие диаграммы классов должны быть модифицированы явным описанием этих классов.

Следует помнить, что на диаграмме кооперации изображаются только те объекты, которые непосредственно в ней участвуют. При этом объекты могут выступать в различных ролях, которые должны быть явно указаны на соответствующих концах связей диаграммы. Применение стереотипов унифицирует кооперацию, обеспечивая ее адекватную интерпретацию как со стороны заказчиков, так и со стороны разработчиков. Тем не менее, целесообразно различать терминологию, используемую на диаграммах кооперации уровня спецификации и уровня примеров.

Так, при построении диаграмм кооперации уровня спецификации желательно применять наиболее понятную заказчику терминологию, избегая технических фраз и словосочетаний. Например, «оформить заказ», «отгрузить товар», «представить отчет», «разработать план» и т. д. Такие известные разработчикам слова как «сервер», «защищенный протокол», «закрытая операция класса», а также стереотипы и помеченные значения на этом уровне применять не рекомендуется. На уровне спецификации нужно стремиться достичь по возможности полного взаимопонимания между заказчиком и командой разработчиков всех вариантов использования проектируемой системы в контексте их кооперации.

При построении диаграмм кооперации уровня примеров терминология должна наиболее точно отражать все аспекты реализации соответствующих объектов и связей. Поскольку диаграмма этого уровня является документацией для разработчиков системы, здесь допустимо использовать весь арсенал стереотипов, ограничений и помеченных значений, который имеется в языке UML. Если типовых обозначений недостаточно, разработчики могут дополнить диаграмму собственными элементами, используя механизм расширений языка UML.

Процесс построения диаграммы кооперации уровня примеров должен быть согласован с процессами построения диаграммы классов и диаграммы последовательности. В первом случае, как уже отмечалось, необходимо следить за использованием только тех объектов, для которых определены порождающие их классы. Во втором случае нужно согласовывать последовательности передаваемых сообщений. Речь идет о том, что не допускается различный порядок следования сообщений для моделирования одного и того же взаимодействия на диаграмме кооперации и диаграмме последовательности. Таким образом, диаграмма кооперации, с одной стороны, обеспечивает концептуально согласованный переход от статической модели диаграммы классов к динамическим моделям поведения, представляемым диаграммами последовательности, состояний и деятельности. С другой стороны, диаграмма этого типа предопределяет особенности реализации модели на диаграммах компонентов и развертывания.

9. РЕКОМЕНДУЕМАЯ УЧЕБНАЯ, СПРАВОЧНАЯ И МЕТОДИЧЕСКАЯ ЛИТЕРАТУРА

1. Пакет прикладных программ Microsoft Office 2003 (или более новой версии) для представления лекционного материала и выполнения лабораторных работ.
2. Меняев, М.Ф. Информационные системы управления предприятием. Часть 1. [Электронный ресурс] – М.: МГТУ им. Н.Э. Баумана, 2012. – 65 с. – Режим доступа: <https://e.lanbook.com/book/52428>
3. Меняев, М.Ф. Информационные системы управления предприятием. Часть 2. [Электронный ресурс] – М.: МГТУ им. Н.Э. Баумана, 2013. – 67 с. – Режим доступа: <http://e.lanbook.com/book/52441>
4. Знаменский, Д.Ю. Информационно-аналитические системы и технологии в государственном и муниципальном управлении. [Электронный ресурс] – СПб: ИЦ Интермедия, 2014. – 180 с.
5. Информационное общество, электронный научный журнал [Электронный ресурс]. Режим доступа: <http://infosoc.iis.ru/>
6. Информационные решения [электронный ресурс]. Режим доступа: Microsoft Solutions Framework, <http://www.microsoft.com/techriet/solutionaccelerators/msf/default.aspx>
7. Информационные решения [электронный ресурс]. Режим доступа: <http://it-management.ru/>
8. Васильев Р.Б. Стратегическое управление информационными системами (Цикл лекций). [электронный ресурс]. Режим доступа: http://www.youtube.com/watch?v=YKxk_1BQwWA
9. Взаимодействие компании-заказчика с компанией-аутсорсером в ИТ сфере. [электронный ресурс]. Режим доступа: <http://www.red-case.ru/>
10. Зеленков Ю.А. Директор по ИТ, к.ф.-м.н. Презентация: НПО «Сатурн»: Эффективная организация ИТ службы крупной компании. [электронный ресурс]. Режим доступа: <http://www.myshared.ru/slide/75107/>
11. <http://e-library.ru> Научная библиотека eLibrary.ru (из любой точки доступа локальной сети КНИТУ-КАИ)
12. Библиотека успешных ИТ-кейсов: Внедрение компанией "Интеллект-Сервис" отраслевого решения для автодилеров incadea на базе Microsoft Dynamics NAV в ЗАО "УралФрансАвто". [электронный ресурс]. Режим доступа: http://raexpert.ru/database/it_cases/?case=200
13. Кейс: модель «ИТ, как сервис» индийской компании Tata Consultancy Services. [электронный ресурс]. Режим доступа: <http://smartsourcing.ru/blogs/keysy/319>
14. Сайт Федеральной службы государственной статистики – <http://www.gks.ru>