МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

**Федеральное государственное автономное образовательное учреждение**

**высшего образования**

**«Национальный исследовательский нижегородский университет им. Н.И. Лобачевского»**

**И.Д. Камскова**

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ по выполнению практических заданий по МДК.01.02 Методы и средства проектирования информационных систем**

Учебно-методическое пособие

Рекомендовано методической комиссией института экономики и предпринимательства ННГУ для обучающихся по специальности среднего профессионального образования 09.02.04 «Информационные системы (по отраслям)»

2017

УДК 681.3

ББК 32.81

Камскова И.Д. Методические рекомендации по выполнению практических заданий по МДК.01.02 «Методы и средства проектирования информационных систем». Учебно-методическое пособие. - Нижний Новгород: Нижегородский государственный университет им. Н.И. Лобачевского, 2017. − 56 с.

Рецензент: к.т.н., доцент Штанюк А.А.

Методические рекомендации направлены на выполнение практической работы по МДК.01.02 «Методы и средства проектирования информационных систем» для учащихся CПО по специальности 09.02.04 «Информационные системы (по отраслям)».

Главной задачей при выполнении практических заданий является получение практических навыков проектирования программного обеспечения информационных систем при помощи различных средств проектирования и используя различные методы проектирования.

Методические рекомендации по  выполнению практической работы разработаны в соответствии с программой по профессиональному модулю ПМ.1 «Эксплуатация и модификация информационных систем».

Ответственный за выпуск:

председатель методической комиссии ИЭП ННГУ Летягина Е.Н.

УДК 681.3

ББК32.81

© **Национальный исследовательский**

**Нижегородский государственный**

**Университет им. Н.И. Лобачевского, 2017**

Оглавление

[1 ПОЯСНИТЕЛЬНАЯ ЗАПИСКА 4](#_Toc512504255)

[2 ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ 6](#_Toc512504256)

[3 МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ВЫПОЛНЕНИЮ практических ЗАДАНИЙ по МДК 01.02 «Методы и средства проектирования информационных систем» 7](#_Toc512504257)

[3.1 Раздел 1. Введение в проектирование ПО 7](#_Toc512504258)

[3.2 Раздел 2. Структурные методы проектирования информационных систем 12](#_Toc512504259)

[3.3 Раздел 3. Метод Сущность-связь (ER) 28](#_Toc512504260)

[3.4 Раздел 4. Визуальное моделирование на языке UML 35](#_Toc512504261)

[3.5 Раздел 5. Технологии создания ПО 50](#_Toc512504262)

[3.6 Раздел 6. Объектно-ориентированное проектирования программного обеспечения 51](#_Toc512504263)

[4 Литература для учащегося 55](#_Toc512504264)

# ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Данные методические рекомендации направлены на выполнение практической работы по МДК.01.02 «Методы и средства проектирования информационных систем» для учащихся CПО по специальности 09.02.04 «Информационные системы (по отраслям)».

Практическая работа студента является одним из основных способов приобретения навыков проектирования программного обеспечения информационных систем при помощи различных средств проектирования и используя различные методы проектирования. Главной задачей практической работы является развитие общих и профессиональных компетенций, умений приобретать знания путем выполнения практических заданий, формирование активного интереса и творческого подхода к решению задач.

Методические рекомендации разработаны в соответствии с программой профессионального модуля ПМ.01 «Эксплуатация и модификация информационных систем» для учащихся CПО по специальности 09.02.04 «Информационные системы (по отраслям)».

В ходе изучения дисциплины ставиться задача формирования следующих компетенций:

**общие:**

ОК 1. Понимать сущность и социальную значимость своей будущей профессии, проявлять к ней устойчивый интерес

ОК 2. Организовывать собственную деятельность, выбирать типовые методы и способы выполнения профессиональных задач, оценивать их эффективность и качество

ОК 3. Принимать решения в стандартных и нестандартных ситуациях и нести за них ответственность

ОК 4. Осуществлять поиск и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития

ОК 5. Использовать информационно - коммуникационные технологии в профессиональной деятельности

ОК 6. Работать в коллективе и команде, эффективно общаться с коллегами, руководством, потребителями

ОК 7. Брать на себя ответственность за работу членов команды (подчинённых), результат выполнения заданий

ОК 8. Самостоятельно определять задачи профессионального и личностного развития, заниматься самообразованием, осознанно планировать повышение квалификации

ОК 9. Ориентироваться в условиях частой смены технологий в профессиональной деятельности

**профессиональные:**

ПК1.1. Собирать данные для анализа использования и функционирования информационной системы, участвовать в составлении отчетной документации, принимать участие в разработке проектной документации на модификацию информационной системы.

ПК1.2. Взаимодействовать со специалистами смежного профиля при разработке методов, средств и технологий применения объектов профессиональной деятельности.

ПК1.3. Производить модификацию отдельных модулей информационной системы в соответствии с рабочим заданием, документировать произведенные изменения.

ПК1.4. Участвовать в экспериментальном тестировании информационной системы на этапе опытной эксплуатации, фиксировать выявленные ошибки кодирования в разрабатываемых модулях информационной системы.

ПК1.5. Разрабатывать фрагменты документации по эксплуатации информационной системы.

ПК1.6. Участвовать в оценке качества и экономической эффективности информационной системы.

ПК1.7. Производить инсталляцию и настройку информационной системы в рамках своей компетенции, документировать результаты работ.

ПК1.8. Консультировать пользователей информационной системы и разрабатывать фрагменты методики обучения пользователей информационной системы.

ПК1.9. Выполнять регламенты по обновлению, техническому сопровождению и восстановлению данных информационной системы, работать с технической документацией.

ПК1.10. Обеспечивать организацию доступа пользователей информационной системы в рамках своей компетенции.

# ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Раздел / тема учебной дисциплины** | **Кол-во часов практической работы** | | | | | |
| 2017 | 2016 | 2015 | | 2014 | |
| **Раздел 1. Введение в проектирование ПО** | | | | | | |
| Тема 1.1. Введение в проектирование ПО | 2 | 2 | 2 | | 2 | |
| Тема 1.2. Методы и технологии проектирования | 4 | 2 | 2 | | 2 | |
| Тема 1.3. Средства проектирования ПО | 4 | 2 | 2 | | 2 | |
| **Раздел 2. Структурные методы проектирования информационных систем** | | | | | | |
| Тема 2.1.  Проведение предпроектного обследования предприятия | 6 | 4 | 4 | | 4 | |
| Тема 2.2. Структурные методы проектирования информационных систем | 16 | 16 | 16 | | 16 | |
| **Раздел 3. Метод Сущность-связь (ER)** | | | | | | |
| Тема 3.1.  Метод Сущность-связь (ER) | 18 | 18 | 18 | | 18 | |
| **Раздел 4. Визуальное моделирование на языке UML** | | | | | | |
| Тема 4.1.  Визуальное моделирование на языке UML | 16 | 16 | 16 | | 16 | |
| **Раздел 5. Технологии создания ПО** | | | | | | |
| Тема 5.1. Жизненный цикл ПО | 4 | 2 | 2 | | 2 | |
| Тема 5.2. Технологии создания ПО | 4 | 4 | 4 | | 4 | |
| **Раздел 6. Объектно-ориентированное проектирование программного обеспечения** | | | | | | |
| Тема 6.1. Моделирование прецедентов | 6 | 6 | 6 | | 6 | |
| Тема 6.2. Проектирование пользовательского интерфейса | 6 | 6 | 6 | | 6 | |
| Тема 6.3. Проектирование баз данных информационных систем | 4 | 4 | 4 | | 4 | |
| Тема 6.4. Проектирование классов | 6 | 6 | 6 | | 6 | |
| **Итого** | **96** | **88** | **88** | **88** | |

# МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ПО ВЫПОЛНЕНИЮ практических ЗАДАНИЙ по МДК 01.02 «Методы и средства проектирования информационных систем»

## Раздел 1. Введение в проектирование ПО

**Тема 1.1. Введение в проектирование ПО**

**Задание 1.1. Выявление командных ролей при работе в группе разработчиков программного обеспечения.**

Определить естественные для разработчика ПО роли в группе, а также те роли, от выполнения которых он предпочел бы отказаться.

**Методические рекомендации**

В каждом из семи блоков данного опросника **распределите 10 баллов** между возможными ответами согласно тому, как вы полагаете они лучше всего подходят вашему собственному поведению. Если вы согласны с каким-либо утверждением на все 100%, вы можете отдать ему все 10 баллов, а остальным не присвоить ничего или распределить все баллы поровну. При этом одному предложению можно присвоить минимум 2 балла.

Проверьте, чтобы сумма всех баллов по каждому блоку не превышала 10.

**Вопросы к тесту Белбина.**

**Блок 1. Какой вклад я могу внести в работу команды:**

10. Я думаю, что способен быстро замечать новые возможности и извлекать из них выгоды.

11. Я могу успешно работать с самыми разными людьми.

12. Генерация идей — моё врожденное достоинство.

13. Моим достоинством является умение находить людей, способных принести пользу команде.

14. Моя способность доводить всё до конца во многом обеспечила мою профессиональную эффективность.

15. Я готов перенести временную непопулярность, если вижу, что мои действия принесут в конечном счете полезные результаты.

16. Я быстро выясняю, что сработает в данной ситуации, если в подобную ситуацию я уже попадал.

17. Личные заблуждения и предубеждения не мешают мне находить и доказывать преимущества альтернативных действий.

**Блок 2. Мои недостатки, которые могут проявиться в командной работе:**

20. Я чувствую себя неуверенно на совещании, если отсутствуют четкая повестка дня и контроль за её соблюдением.

21. Я склонен быть слишком великодушным к людям, имеющим правильную точку зрения, но не высказывающим её открыто.

22. Я склонен слишком много говорить, когда в группе обсуждаются новые идеи.

23. Вследствие моей осмотрительности я не склонен быстро и с энтузиазмом присоединяться к мнению коллег.

24. Я иногда выгляжу авторитарным и нетерпимым, когда чувствую необходимость достичь чего-то.

25. Мне трудно повести людей за собой, поскольку я слишком подвержен влиянию атмосферы, царящей в группе.

26. Я слишком захвачен идеями, которые мне приходят в голову, и поэтому плохо слежу за тем, что происходит вокруг.

27. Мои коллеги находят, что я слишком много внимания уделяю деталям и чрезмерно беспокоюсь о том, что дела идут неправильно.

**Блок 3. Участие в совместном проекте:**

30. Я умею влиять на людей, не оказывая на них давления.

31. Врожденная осмотрительность предохраняет меня от ошибок, возникающих из-за невнимательности.

32. Я готов оказать давление, чтобы совещание не превращалось в пустую трату времени и не терялась из виду основная цель обсуждения.

33. Можно рассчитывать на поступление от меня оригинальных предложений.

34. Я всегда готов поддержать любое предложение, если оно служит общим интересам.

35. Я энергично ищу среди новых идей и разработок свежайшие.

36. Я надеюсь, что моя способность выносить беспристрастные суждения признаётся всеми, кто меня знает.

37. На меня можно возложить обязанности следить за тем, чтобы наиболее существенная работа была организована должным образом.

**Блок 4. Особенности моего стиля работы в команде:**

40. Я постоянно стараюсь лучше узнать своих коллег.

41. Я неохотно возражаю своим коллегам и не люблю сам быть в меньшинстве.

42. Я обычно нахожу вескую аргументацию против плохих предложений.

43. Я полагаю, что обладаю талантом быстро организовать исполнение одобренных планов.

44. Я обладаю способностью избегать очевидных решений и умею находить неожиданные.

45. Я стремлюсь добиться совершенства при исполнении любой роли в командной работе.

46. Я умею устанавливать контакты с внешним окружением команды.

47. Я способен воспринимать любые высказываемые мнения, но без колебаний подчиняюсь мнению большинства после принятия решения.

**Блок 5. Я получаю удовлетворение от работы, потому что:**

50. Мне доставляет удовольствие анализ ситуаций и взвешивание всех шансов.

51. Мне нравится находить практические решения проблем.

52. Мне нравиться сознавать, что я создаю хорошие рабочие взаимоотношения.

53. Я способен оказывать сильное влияние на принятие решений.

54. Я получаю возможность встретиться с людьми, способными предложить что-то новое для меня.

55. Я способен добиться согласия людей на реализацию необходимого курса действий.

56. Я чувствую себя в своей стихии, когда могу уделить задаче все мое внимание.

57. Мне нравится находить задачи, требующие напряжения воображения.

**Блок 6. Если мне неожиданно предложат решить трудную задачу за ограниченное время с незнакомыми людьми, то:**

60. Я бы почувствовал необходимость сначала в одиночестве обдумать пути выхода из тупика, прежде чем начать действовать.

61. Я был бы готов работать с человеком, указавшим наиболее позитивный подход, каковы бы ни были связанные с этим трудности.

62. Я бы попытался найти способ разбиения задачи на части в соответствии с тем, что лучше всего умеют делать отдельные члены команды.

63. Присущая мне обязательность помогла бы нам не отстать от графика.

64. Я надеюсь, мне бы удалось сохранить хладнокровие и способность логически мыслить.

65. Я бы упорно добивался достижения цели, несмотря ни на какие помехи.

66. Я был бы готов действовать силой положительного примера при появлении признаков отсутствия прогресса в командной работе.

67. Я бы организовал дискуссию, чтобы стимулировать выдвижение новых идей и придать начальный импульс командной работе.

**Блок 7. Проблемы, с которыми я сталкиваюсь, работая в команде:**

70. Я склонен проявлять нетерпимость по отношению к людям, мешающим, по моему мнению, прогрессу в делах группы.

71. Окружающие иногда критикуют меня за чрезмерный рационализм и неспособность к интуитивным решениям.

72. Мое стремление обеспечить условия, чтобы работа выполнялась правильно, может приводить к снижению темпов.

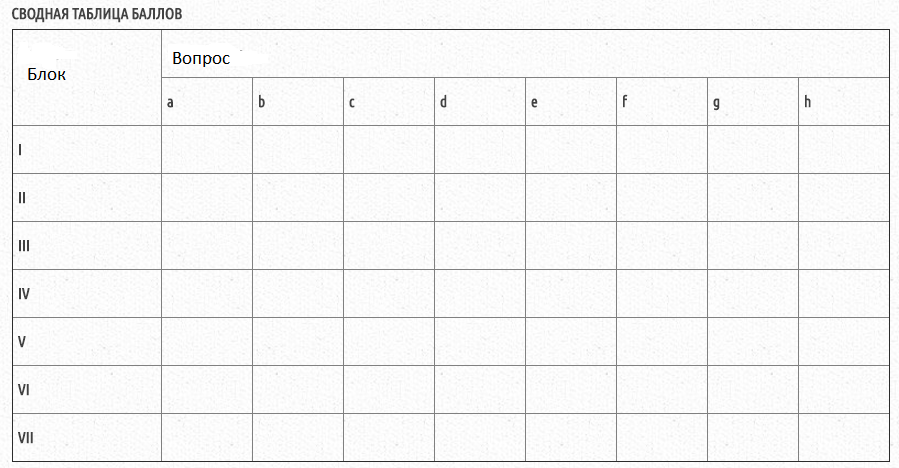
73. Я слишком быстро утрачиваю энтузиазм и стараюсь почерпнуть его у наиболее активных членов группы.

74. Я тяжел на подъем, если не имею ясных целей.

75. Мне иногда бывает очень трудно разобраться во встретившихся мне сложностях.

76. Я стесняюсь обратиться за помощью к другим, когда не могу что-либо сделать сам.

77. Я испытываю затруднения при обосновании своей точки зрения, когда сталкиваюсь с серьезными возражениями.  
**Расшифровка результатов**

Перенесите свои баллы из каждого блока опросника в таблицу внизу. Проследите, чтобы общая сумма всех баллов в итоговой строке была равна 70. Если итог не равен 70, пересчитайте, пожалуйста, еще раз, где-то была допущена ошибка.  


Затем суммируйте баллы в каждом столбце для получения общей оценки командной роли. Наивысший балл по командной роли показывает, насколько хорошо Вы можете исполнять эту роль в управленческой команде. Следующий результат после наивысшего может обозначать поддерживающую роль, на которую Вы можете переключиться, если Ваша основная командная роль по каким-либо причинам не нужна группе. Два самых низких балла по командной роли выявляют Ваши возможные «слабые места».

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Реализатор | Координатор | Творец | Мыслитель | Исследователь | Эксперт | Дипломат | Исполнитель |

Определяют восемь ролей, которые, если все они присутствуют в команде, дают ей больше шансов на успех. Вот эти роли:

Координатор (председатель) группы уточняет цели, определяет повестку дня, определяет приоритеты, выделяет проблемы, подводит итоги, имеет решающее значение, но не доминирует в дискуссии.

Творец (формирователь) придает форму коллективным усилиям, глядя в суть дискуссии и практических соображений относительно целесообразности определенного проекта. Может давить на команду, но это дает некоторые результаты.

Мыслитель (генератор идей) является источником оригинальных идей, пожеланий и предложений, которые, как правило, оригинальны и радикальны.

Эксперт (оценщик) измеряет и беспристрастно анализирует, а с помощью объективности, останавливает команду и берет на себя обязательство по определению ошибочной задачи.

Исполнитель (работник) преобразует решения и стратегии в определенные и выполнимые задачи, логически рассматривая цели.

Исследователь (разведчик) выходит за пределы команды, принося идеи, информацию и события. Он продавец, дипломат, коммуникатор и исследователь.

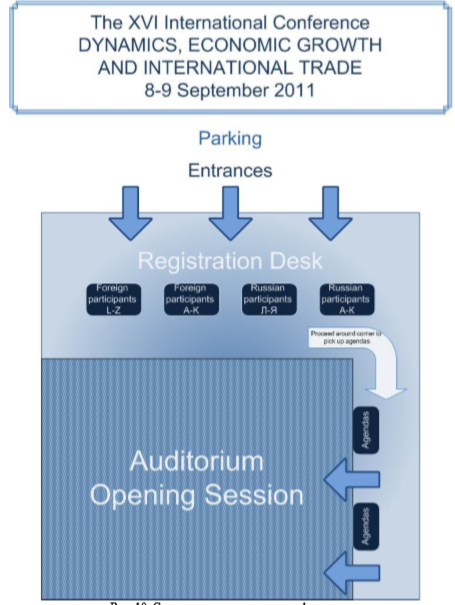
Дипломат (коллективист) способствует сохранению целостности команды, предотвращает ее распад в случае давления либо стресса

Реализатор (доводчик) имеет постоянное чувство сроков.

**Тема 1.2. Методы и технологии проектирования**

**Задание 1.2. Создайте схему места проведения конференции согласно рисунку в графическом редакторе Microsoft Visio.**

**Цель задания:** Научиться работать в редакторе **Microsoft Visio.**

****

**Методические рекомендации**

В наборах элементов Visio содержатся коллекции фигур. Фигуры в каждом наборе элементов имеют схожие черты. В этих наборах могут содержаться коллекции фигур для создания определенного типа диаграммы или несколько различных видов одной и той же фигуры. Чтобы развернуть набор элементов, следует щелкнуть по его заголовку.

Для просмотра и выбора других фигур можно либо в меню Файл выбрать элемент Фигуры и перейти вниз до набора элементов, который необходимо открыть, либо воспользоваться кнопкой Фигуры на стандартной панели инструментов.

Работая с Visio, вы будете работать с различными фигурами, которые являются элементарными составляющими самых разнообразных схем. Фигуры в Visio обладают интерактивным поведением, то есть определенными образом реагируют на выполняемые с ними действия. В Visio различают два типа фигур: одномерные и двухмерные. От типа фигуры зависит ее поведение.

Одномерная фигура имеет начальную и конечную точку, основное ее свойство это соединять две другие фигуры. Двухмерные фигуры выделяются с помощью восьми зеленых маркеров выделения и используются для представления специальных объектов или идей. У некоторых фигур в Visio имеется дополнительное поведение, которое характеризует именно данный тип фигур и обозначается желтыми управляющими маркерами.

Для вставки стандартных фигур выбираем нужную фигуру из набора элементов и перетаскиваем ее на страницу. Размеры можно изменить с помощью маркеров. Наборы элементов, отображаемые в окне

Фигуры, зависят от выбранного шаблона. В случае, когда известно название необходимого элемента, искать фигуры можно с помощью окна Поиск фигур. Тогда будут найдены фигуры из всех шаблонов. Также фигуры могут быть созданы самостоятельно или получены от других пользователей.

В контекстном меню для фигуры с помощью меню Формат можно поменять цвет и формат линий, тень и заливку фигуры. Для группировки объектов нужно выделить необходимые объекты общей рамкой, и выбрать команду Фигура/Группировать.

Чтобы переместить фигуру выполните следующие действия: поместите указатель на фигуру. Когда указатель сменится на четырехстороннюю стрелку, перетащите фигуру, удерживая нажатой кнопку мыши.

Чтобы выровнять фигуру относительно другой фигуры, используйте линии динамической сетки. (в меню Сервис выберите команду Привязать и приклеить, на вкладке Общие установите флажок Динамическая сетка и нажмите кнопку ОК) Чтобы немного подвинуть фигуру, выделите ее и нажмите клавишу со стрелкой. Чтобы подвинуть фигуру точно на один пиксель, нажмите клавишу со стрелкой, удерживая нажатой клавишу SHIFT.

Чтобы скопировать фигуру, нужно ее выделить. Затем, удерживая нажатой клавишу CTRL, перетащите копию фигуры в нужное место. Отпустите кнопку мыши. Поворот одномерных фигур выполняется с использованием начальной или конечной точки. Для поворота двумерных фигур следует перемещать маркеры поворота зеленого цвета. Также для поворота и отражения фигур можно использовать пункт Повернуть/Отразить в меню Фигура.

Для добавления фона используется набор элементов Фоновые рисунки. Если он не открыт, то выберите в меню Файл команду Фигуры, затем – команду Дополнительные решения Visio, а после этого – команду Фоновые рисунки. Перетащите фон на страницу. При перемещении будет создана новая страница Фон Visio-1, на которой можно редактировать фон, добавлять логотип или другую информацию. При добавлении новой страницы по умолчанию будет указан выбранный фон. Чтобы удалить фон страницы, следует перетащить фигуру Без фона. Для добавления рамок следует воспользоваться набор Рамки и заголовки. В рамки также можно вписывать текст. Можно добавлять заголовок отдельно от рамки.

С помощью текста в Visio можно помечать фигуры и соединители, создавать заголовки схем, создавать отдельные надписи. Для добавления текста в фигуру (внедренный текст) следует щелкнуть по фигуре левой кнопкой мыши и ввести текст. Масштаб фигуры в Visio будет увеличен. По завершении работы щелкните в пустой области страницы рисунка, чтобы вернуться к прежнему масштабу. На фигуре появится текст. Для редактирования текста следует щелкнуть по фигуре дважды. Для добавления текста в группу можно воспользоваться кнопкой Текст на стандартной панели инструментов. С помощью этой кнопки можно также создать текстовую фигуру (надпись) в любом месте листа.

Чтобы программа Visio не увеличивала текст, выберите в меню Сервис команду Параметры и перейдите на вкладку Общие. В области Параметры окна документа введите для размера шрифта значение 0 (ноль) в поле Автоматически увеличивать масштаб текста при редактировании текста мельче. Текстовый блок – это текстовая область, связанная с фигурой. Это самостоятельная часть фигуры, присоединенная к ней. Для управления текстовым блоком необходимо сначала выбрать инструмент Блок текста (Он находится в списке кнопки Текст). Затем его можно использовать для перемещения, поворота и изменения размера текстового блока.

Чтобы размер текста изменялся одновременно с размером фигуры, нужно выполнить следующие действия (возможно не для всех фигур): выделить фигуру с текстом; в меню Окно выбрать пункт Показать таблицу свойств фигуры; в открывшемся окне найти раздел Знак. В ячейке Размер удалите содержимое ячейки и введите HEIGHT/5. Закройте лист.

Для установки и снятия запрета на изменения фигуры следует выделить фигуру, выбрать в меню Формат команду Защита и выбрать те атрибуты фигуры, для которых нужно установить блокировку, или снять флажки с тех атрибутов, которые необходимо разблокировать, после чего нажать кнопку ОК. Чтобы защита начала действовать, нужно в меню Вид выбрать команду Окно проводника по документам, щелкнуть правой кнопкой мыши имя документа, а затем в контекстном меню выбрать команду Защитить документ, после чего в группе Защитить установить флажок Фигуры и нажать кнопку ОК.

**Тема 1.3. Средства проектирования ПО**

**Задание 1.3. Анализ предметной области**

**Цели задания**

1. Научиться работать в команде
2. Ознакомиться с элементами анализа предметной области

**Задание**

1. Придумать, продумать и нарисовать структуру организации, для которой будет создавать автоматизированная информационная система.
2. Описать схему взаимодействия организации с внешней средой.
3. Провести анализ входных и выходных информационных потоков и соответствующих им документов.

**Результат работы**

В результате работы должен быть предоставлен отчет (оформление и т.д. полностью зависит от команды). При проектировании схемы рекомендуется использовать Microsoft Visio или Microsoft Word

**Варианты предметных областей**

1. Система бронирования турпутевок
2. Управление взаимоотношениями с клиентами
3. Система управления рестораном
4. Система автоматизации аптеки
5. Система автоматизации гостиницы.
6. Система кредитования банка
7. Библиотечная система
8. Система учета в фитнес-центре.
9. Система автоматизации речных круизов
10. Автопрокат

## Раздел 2. Структурные методы проектирования информационных систем

**Тема 2.1.  Проведение предпроектного обследования предприятия**

**Цель выполнения задания** - получение практических навыков предпроектного обследования предприятия.

**Задание** - Используя материалы лекции, рекомендованную литературу, провести необходимое предпроектное обследование предприятия, выбранного в теме 1.3. Обследование делается только в рамках задания.

**Результатом** выполнения заданий является текстовый документ «Отчет об экспресс-обследовании предприятия «…».

**Методические рекомендации**

Сбор информации для построения полной бизнес-модели организации часто сводится к изучению информационных потоков и функций подразделений, а также производится путем интервьюирования и анкетирования. К началу работ по обследованию организация обычно предоставляет комплект документов, в состав которого обычно входят:

1. Сводная информация о деятельности предприятия.
   1. Информация об управленческой, финансово-экономической, производственной деятельности предприятия.
   2. Сведения об учетной политике и отчетности.
2. Регулярный документооборот предприятия.
   1. Реестр входящей информации.
   2. Реестр внутренней информации.
   3. Реестр исходящей информации.
3. Сведения об информационно–вычислительной инфраструктуре предприятия.
4. Сведения об ответственных лицах.

Результатом предпроектного обследования является "Отчет об экспресс-обследовании предприятия", структура которого приведена ниже.

1. Краткое схематичное описание бизнес-процессов:
   * управление закупками и запасами;
   * управление производством;
   * управление продажами;
   * управление финансовыми ресурсами.
2. Основные требования и приоритеты автоматизации.
3. Оценка необходимых для обеспечения проекта ресурсов заказчика.
4. Оценка возможности автоматизации, предложения по созданию автоматизированной системы с оценкой примерных сроков и стоимости.

Документы, входящие в отчет об обследовании, могут быть представлены в виде текстового описания или таблиц, примерная форма которых приведена ниже.

Операции бизнес-процесса

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Операция | Исполнитель | Как часто | Входящие документы (документы-основания) | Исходящий документ (составляемый документ) |

Описание документов бизнес-процесса

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Составляемый документ (исходящий документ) | Операция | Кто составляет (исполнитель) | Как часто | Документы-основания (входящие документы) |

**Критерии оценки задания:**

Для получения положительной оценки по данному заданию необходимо:

1. Грамотно выделить существующие бизнес-процессы на предприятии.

2. Грамотно выделить документы каждого бизнес-процесса

3. Уметь составлять отчет об обследовании предприятия.

**Тема 2.2. Структурные методы проектирования информационных систем**

**Задание 2.2. Проектирование модели IDEF0 средствами AllFusion Process Modeler (BPwin)**

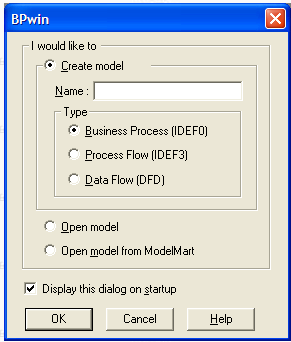
**Методические рекомендации**

AllFusion Process Modeler (далее BPwin) — CASE-средство для моделирования бизнес-процессов, позволяющая создавать диаграммы в нотации IDEF0, IDEF3, DFD.

В процессе моделирования BPwin позволяет переключиться с нотации IDEF0 на любой ветви модели на нотацию IDEF3 или DFD и создать смешанную модель. BPwin поддерживает функционально-стоимостной анализ (ABC).

1.1 Для запуска средства проектирования выберите последовательно Пуск-Все программы-Computer Associates - AllFusion Process Modeler – Bpwin.

1.2 Работа с программой начинается с создания новой модели, для которой нужно указать имя и тип



Как видно из рисунка, можно начать создавать разные модели IDEF0, IDEF3, DFD.

Если выбрать тип *Business Process (IDEF0),* то в созданной модели можно производить декомпозицию работ в нотациях IDEF0, IDEF3 и DFD; если выбран тип *Data Flow (DFD)* — в нотациях DFD и IDEF3; если же выбран тип *Process Flow (IDEF3) —* то только в нотации IDEF3.

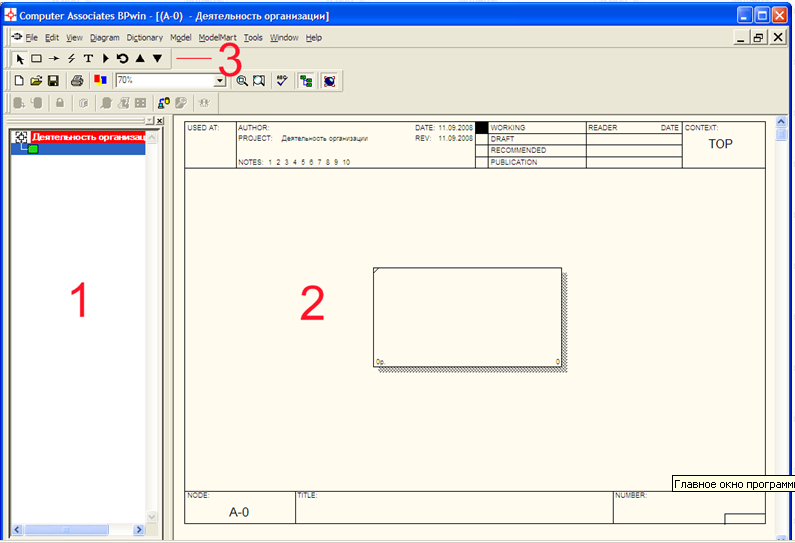
Мы будем создавать модель IDEF0 и назовем ее **QUILL**

После ввода имени модели и выбора ее типа BPWin сразу предложит задать параметры модели

1.3 Введите на вкладке General в поля Author и Author initials свою фамилию и инициалы в латинской транскрипции. Остальные вкладки имеют следующие назначения:

* *General*— автор модели и его инициалы;
* *Numbering —* формат нумерации работ и диаграмм и порядок ее отображения на диаграммах;
* *Display —* список элементов отображения на диаграммах;
* *Layout —* параметры расположения;
* *ABC Units —* единицы функционально-стоимостного анализа;
* *Page Setup —* параметры страницы;
* *Header/Footer* — параметры верхнего и нижнего колонтитула.

После задания свойств модели появляется главное окно программы, состоящее из трех основных частей:

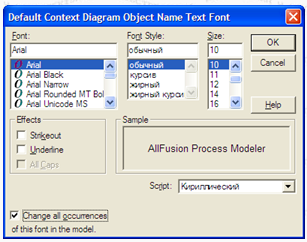


1 - обозреватель модели (Model Explorer) — отображает структуру модели (имеющиеся диаграммы и их иерархию);

2 - основная часть — в ней отображаются диаграммы, с которыми ведется работа;

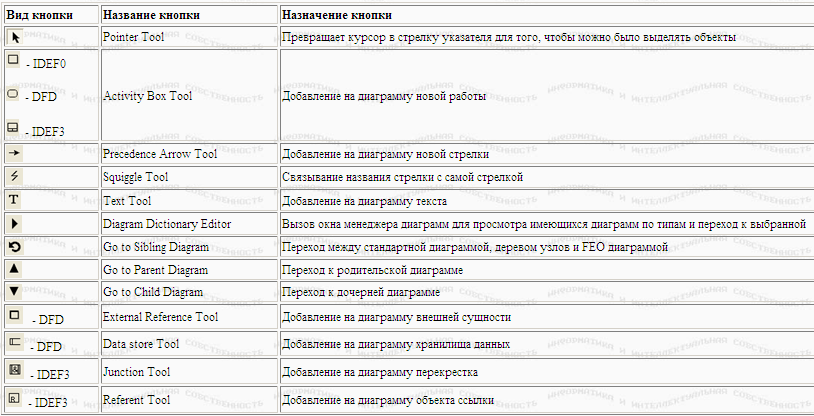
3 - панели инструментов, из которых наибольший интерес представляет панель инструментов Model Toolbox.

**Примечание**. В созданной модели с настройками по умолчанию некорректно отображаются русские символы. Чтобы устранить этот недостаток, необходимо подкорректировать используемые в модели шрифты. Для этого в меню **Model -> Default Fonts** необходимо последовательно пройтись по всем пунктам, выбрать в выпадающем списке **Script** значение кириллический и поставить галочку **Change all occurrences**.



**Панель инструментов *Model Toolbox***

Данная панель инструментов отвечает за создание разнообразных графических элементов модели. В зависимости от типа текущей диаграммы набор кнопок на ней меняется.



1. **Контекстная диаграмма**

Рассмотрим деятельность компании QUILL, которая существует 5 лет и занимается в основном сборкой и продажей настольных компьютеров и ноутбуков. Годовой оборот компании составляет примерно 20 млн долларов. Компания закупает компоненты для компьютеров от трех независимых поставщиков, а не производит компоненты самостоятельно. Она только собирает и тестирует компьютеры. Компания реализует продукцию через магазины и специализируется на покупателях, для которых главный критерий при покупке - стоимость компьютера. Предполагаемый объем рынка для компании QUILL в последующие 2 года - 50 млн долларов.

Несмотря на некоторое увеличение объема продаж, прибыли уменьшаются, растет конкуренция на рынке. Чтобы не потерять позиции, компания решает проанализировать текущие бизнес-процессы и реорганизовать их с целью увеличения эффективности производства и продаж. Основная деятельность компании выглядит следующим образом:

* продавцы принимают заказы клиентов;
* операторы 1 группируют заказы по типам компьютеров;
* операторы 2 собирают и тестируют компьютеры;
* операторы 3 упаковывают компьютеры согласно заказам;
* кладовщик отгружает клиентам заказы.

В настоящее время компания использует купленную бухгалтерскую информационную систему, которая позволяет оформить заказ, счет и отследить платежи по счетам.

Улучшение деятельности компании должно касаться структуры управления компанией, эффективности производства и внутреннего контроля. В результате реорганизация может потребовать внедрения новой корпоративной информационной системы (состоящей не только из бухгалтерского модуля).

2.1 Созданная модель уже содержит **контекстную диаграмму** с единственной работой ("черный ящик") в той нотации, которая была выбрана на этапе создания модели.

Теперь необходимо дать этой работе название и при необходимости задать ее свойства. Для этого нужно вызвать окно свойств работы,  дважды щелкнуть по ней мышью или щелчок правой кнопкой по объекту во вкладке **Activities** позволяет воспользоваться контекстным меню для редактирования его свойств.

Введите название контекстной диаграммы

2.2 На вкладке **Definition** диалогового окна **Activity Properties** в текстовое поле **Definition** (Определение) внесите "Текущие бизнес-процессы компании" . Текстовое поле **Note (**Примечания**)** оставьте незаполненным.

Обратите внимание на кнопку на панели инструментов. Эта кнопка включает и выключает инструмент просмотра и навигации - **Model Explorer** (Браузер модели). **Model Explorer** имеет три вкладки – **Activities** [[egg]], **Diagrams** [[egg]]и **Objects** [[egg]]. Во вкладке **Activities** щелчок правой кнопкой по объекту в браузере модели позволяет выбрать опции редактирования его свойств.

Если вам непонятно, как выполнить то или иное действие, вы можете вызвать контекстную помощь - клавиша **F 1** или воспользоваться меню **Help**.

2.3 Перейдите в меню **Model/Model Properties**. Во вкладке **General** диалогового окна **Model Properties** в текстовое поле **Model name** следует внести имя модели "Деятельность компании", а в текстовое поле **Project -** имя проекта "Модель деятельности компании", и, наконец, в текстовое **Time Frame** (Временной охват) - **AS - IS (** Как есть**)**. (Рисунок)

2.4 Во вкладке **Purpose** диалогового окна **Model Properties** в текстовое поле **Purpose** (цель) внесите данные о цели разработки модели - "Моделировать текущие ( AS - IS ) бизнес-процессы компании", а в текстовое поле **Viewpoint** (точка зрения) - "Директор".

2.5 Во вкладке **Definition** диалогового окна **Model Properties** в текстовое поле **Definition** (Определение) внесите "Это учебная модель, описывающая деятельность компании" и в текстовое поле **Scope** (охват) - "Общее управление бизнесом компании: исследование рынка, закупка компонентов, сборка, тестирование и продажа продуктов".

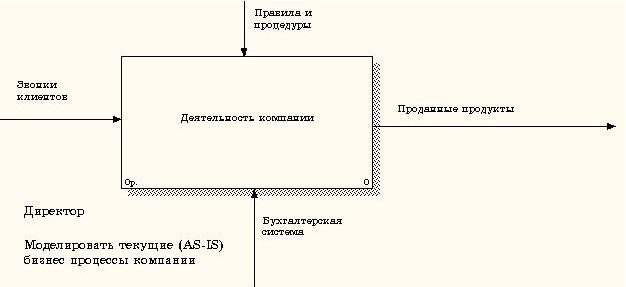
2.6 Далее необходимо разместить на диаграмме стрелки. Для этого следует нажать на *Model Toolbox* кнопку ***Precedence Arrow Tool*** (курсор примет форму крестика со стрелкой), щелкнуть по тому месту, откуда стрелка должна выходить и затем щелкнуть по тому месту, куда стрелка должна заходить (BPwin подсветит эти места при наведении на них курсора).

Для задания названия стрелки нужно нажать на *Model Toolbox* кнопку *Pointer Tool* и затем дважды щелкнуть по стрелке. В появившемся окне *Arrow Properties* название работы  вводится в поле *Arrow Name* или выбирается из списка имеющихся названий стрелок.

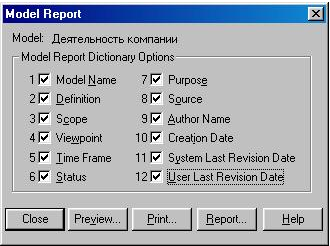
Спроектируйте 4 стрелки контекстной диаграммы

|  |  |  |
| --- | --- | --- |
| **Название стрелки (Arrow Name)** | **Определение стрелки (Arrow Definition )** | **Тип стрелки (Arrow Type )** |
| **Звонки клиентов** | Запросы информации, заказы, техподдержка и т. д. | **Input** |
| **Правила и процедуры** | Правила продаж, инструкции по сборке, процедуры тестирования, критерии производительности и   т. д. | **Control** |
| **Проданные продукты** | Настольные и портативные компьютеры | **Output** |
| **Бухгалтерская система** | Оформление счетов, оплата счетов, работа с заказами | **Mechanism** |

2.7 С помощью кнопки **Text Tool** внесите текст в поле диаграммы - точку зрения и цель Контекстная диаграмма должна принять следующий вид:



2.8 Создайте отчет по модели. В меню **Tools / Reports / Model Report** задайте опции генерирования отчета (установите галочки), как показано на рисунке,



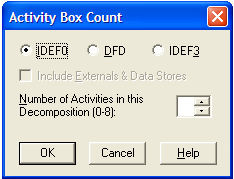
и нажмите кнопку **Preview** (Предварительный просмотр)

2.9 Сохраните проект по именем QUILL.bp1

1. **Создание диаграммы декомпозиции**

После размещения стрелок на диаграмме можно проводить декомпозицию работ. Мы будем декомпозировать контекстную диаграмму, содержащую одну работу. Диаграмма декомпозиции будет содержать дочерние работы, имеющие общую родительскую работу.

3.1 Для декомпозиции работы следует выделить работу, которую нужно декомпозировать, и нажать кнопку ***Go to Child Diagram***. Появится окно, в котором необходимо выбрать в какой нотации проводить декомпозицию и количество дочерних работ.



Мы будем декомпозировать работу в нотации IDEF0 и количество дочерних работ – 3. Нажмите ОК.

Будет создана диаграмма декомпозиции с указанным числом работ.

3.2 Внесите имя, определение, статус и источник для каждой работы согласно данным таблицы.

|  |  |
| --- | --- |
| **Название работы ( Activity Name )** | **Определение работы ( Activity Definition )** |
| Продажи и маркетинг | Телемаркетинг и презентации, выставки |
| Сборка и тестирование компьютеров | Сборка и тестирование настольных и портативных компьютеров |
| Отгрузка и получение | Отгрузка заказов клиентам и получение компонентов от поставщиков |

3.3 Для изменения свойств работ после их внесения в диаграмму можно воспользоваться словарем работ. Вызов словаря производится при помощи пункта главного меню Dictionary / Activity .

Если описать имя и свойства работы в словаре, ее можно будет внести в диаграмму позже с помощью кнопки **Activity Box Tool** в палитре инструментов. Невозможно удалить работу из словаря, если она используется на какой-либо диаграмме. Если работа удаляется из диаграммы, из словаря она не удаляется. Имя и описание такой работы может быть использовано в дальнейшем. Для добавления работы в словарь необходимо перейти в конец списка и щелкнуть правой кнопкой по последней строке. Возникает новая строка, в которой нужно внести имя и свойства работы. Для удаления всех имен работ, не использующихся в модели, щелкните по кнопке **Purge (Чистить)** .

3.4 При декомпозиции работы входящие в нее и исходящие из нее стрелки автоматически появились на диаграмме декомпозиции, но при этом не касаются работ. Такие стрелки называются **несвязанными ( uncorrected border arrow )** и воспринимаются BPWin как синтаксическая ошибка.

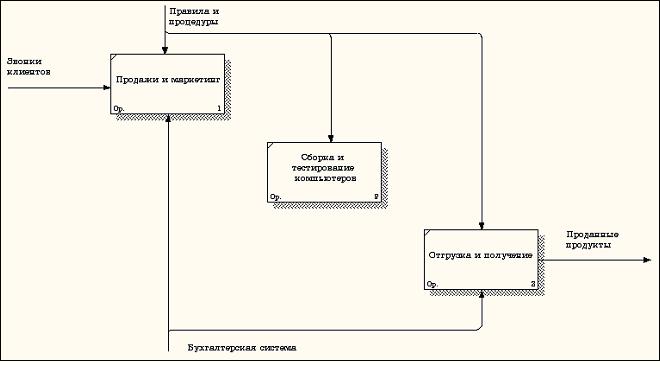
Для ее устранения следует перейти в режим рисования стрелок и связать граничные стрелки с остальными так, как показано на рисунке.

**Для связывания стрелок входа**, управления и механизма необходимо в режиме редактирования стрелки щелкнуть по наконечнику стрелки и щелкнуть по соответствующему сегменту работы.

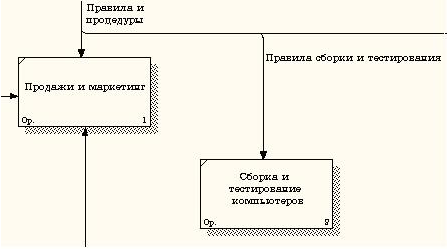
**Для связывания стрелки выхода** необходимо щелкнуть по сегменту выхода работы, а затем по стрелке.

**Для разветвления стрелки** нужно в режиме редактирования стрелки щелкнуть по фрагменту стрелки и щелкнуть по соответствующему сегменту работы.

Перейдите в режим рисования стрелок и свяжите граничные стрелки, воспользовавшись кнопкой Presedence Arrow Tool на палитре инструментов так, как это показано на рисунке.

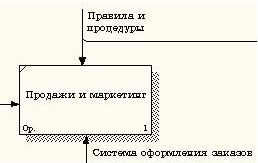


3.5 Правой кнопкой мыши щелкните по ветви стрелки управления работы "Сборка и тестирование компьютеров" и переименуйте ее в "Правила сборки и тестирования".



Внесите определение для новой ветви: "Инструкции по сборке, процедуры тестирования, критерии производительности и т. д."

3.6 Правой кнопкой мыши щелкните по ветви стрелки механизма работы "Продажи и маркетинг" и переименуйте ее как "Система оформления заказов".

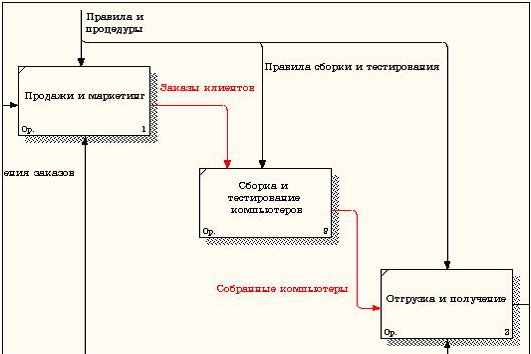


3.7. Альтернативный метод внесения имен и свойств стрелок - использование словаря стрелок (вызов словаря - меню Dictionary / Arrow ). Если внести имя и свойства стрелки в словарь, ее можно будет внести в диаграмму позже.

Стрелку нельзя удалить из словаря, если она используется на какой-либо диаграмме. Если удалить стрелку из диаграммы, из словаря она не удаляется. Имя и описание такой стрелки может быть использовано в дальнейшем.

Для добавления стрелки необходимо перейти в конец списка и щелкнуть правой кнопкой по последней строке. Возникает новая строка, в которой нужно внести имя и свойства стрелки.

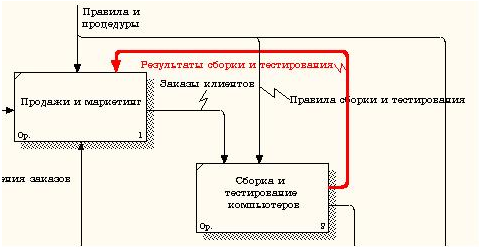
3.8. Создайте новые внутренние стрелки так, как показано на рисунке.



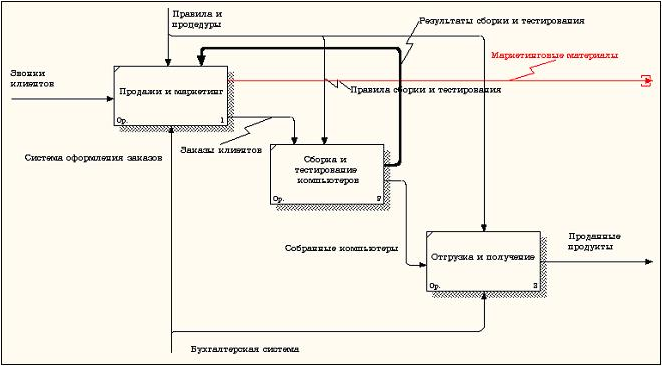
3.9. Создайте стрелку **обратной связи (по управлению)** "Результаты сборки и тестирования", идущую **от** работы "Сборка и тестирование компьютеров" **к** работе "Продажи и маркетинг" .

Измените, при необходимости, стиль стрелки (толщина линий) и установите опцию Extra Arrowhead (Дополнительный Наконечник стрелы) (из контекстного меню).

Методом drag & drop перенесите имена стрелок так, чтобы их было удобнее читать. Если необходимо, установите из контекстного меню Squiggle (Загогулину). Результат возможных изменений показан на рисунке.



3.10. Создайте **новую граничную стрелку выхода** "Маркетинговые материалы", выходящую **из** работы "Продажи и маркетинг". Эта стрелка автоматически не попадает на диаграмму верхнего уровня и имеет квадратные скобки на наконечнике (туннель).



3.11. Щелкните правой кнопкой мыши по квадратным скобкам и выберите пункт меню Arrow Tunnel .

В диалоговом окне Border Arrow Editor (Редактор Граничных Стрелок) выберите опцию Resolve it to Border Arrow ( Разрешить как Граничную Стрелку ) .

Для стрелки "Маркетинговые материалы" выберите опцию Trim (Упорядочить) из контекстного меню. Результат выполнения задания показан на рисунке.



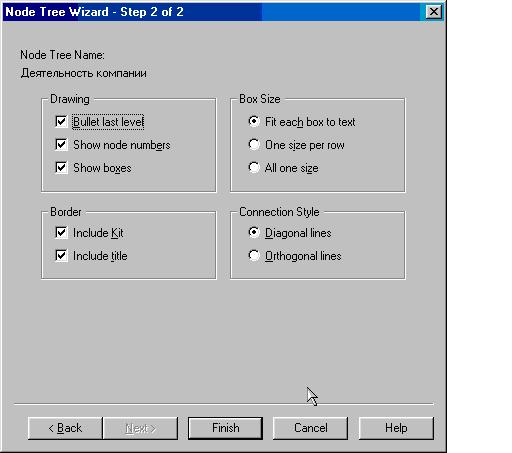
* 1. Сохраните задание

1. **Создание диаграммы узлов и FEO диаграммы**

3.1. Выберите пункт главного меню **Diagram / Add Node Tree**.

3.2. В первом диалоговом окне гида **Node Tree Wizard** внесите имя диаграммы, укажите диаграмму корня дерева и количество уровней как на рисунке.

3.3. Во втором диалоговом окне гида **Node Tree Wizard** установите опции, как показано на рисунке.



3.4. Щелкните по кнопке **Finish**. В результате будет создана диаграмма дерева узлов (**Node tree Diagram )**.

3.5. Диаграмму дерева узлов можно модифицировать. Нижний уровень может быть отображен не в виде списка, а в виде прямоугольников, так же как и верхние уровни. Для модификации диаграммы правой кнопкой мыши щелкните по свободному месту, не занятому объектами, выберите меню **Node tree Diagram Properties** и во вкладке **Style** диалога **Node Tree Properties** отключите опцию **Bullet Last Level**. Щелкните по **ОК**.

1. **Создание FEO-диаграммы**

Предположим, что при обсуждении бизнес-процессов возникла необходимость детально рассмотреть взаимодействие работы **"Сборка и тестирование компьютеров"** с другими работами. Чтобы не портить диаграмму декомпозиции, создайте **FEO**-диаграмму (**FEO** – расшифровывается как «только для экспозиции»), на которой будут только стрелки работы **"Сборка и тестирование компьютеров"**.

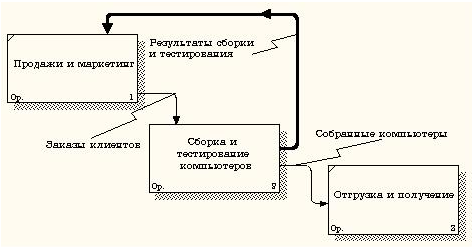
4.1. Выберите пункт главного меню **Diagram / Add FEO Diagram.**

**4.**2. В диалоговом окне **Add New FEO Diagram** выберите тип и внесите имя диаграммы **FEO.**

4.3. Для определения содержания диаграммы перейдите в пункт меню **Diagram / Diagram Properties** и во вкладке **Diagram Text** внесите определение.

4.4. Для определения содержания диаграммы перейдите в пункт меню **Diagram / Diagram Properties** и во вкладке **Diagram Text** внесите определение.

4.5. Удалите лишние стрелки на диаграмме **FEO**. Результат показан на рисунке.



Для перехода между стандартной диаграммой, деревом узлов и **FEO** используйте кнопку **Go to Sibling Diagram** на палитре инструментов.

1. **Создание FEO-диаграммы**

В результате проведения экспертизы от сотрудников производственного отдела получена дополнительная информация - оказалось, что неисправные компоненты направляются на отгрузку. Для уточнения информации необходима дополнительная экспертиза в отделе отгрузки. Требуется создать FEO для проведения такой экспертизы.

Имя новой диаграммы: “**Уточнение по итогам экспертизы**”.

Для этого следует построить FEO на основе диаграммы A0 и добавить стрелку “Неисправные компоненты”, которая идет с выхода работы ‘Сборка и тестирование компьютеров” на вход работы “Отгрузка и получение”.

Древовидные и FEO-диаграммы объединяются под названием “родственные” диаграммы. Они не отражаются непосредственно в дереве модели (**Model Explorer**), однако дерево модели может быть использовано для их открытия. Для этого следует включить Model Explorer, переключиться на вкладку Diagrams, а затем щелкнуть правой кнопкой мыши на названии диаграммы.

Подготовка к практической работе включает в себя изучение конспекта лекций, основной и дополнительной литературы, Internet-источников, выполнение самостоятельного практического задания по предложенным темам.

Ключевым для данной темы являются понятия структурных методов разработки модели бизнес-процессов предприятия.

Студенты должны знать методологии построения моделей IDEF0, IDEF3, DFD. Закрепление навыков проектирования моделей вышеперечисленными методами осуществляется при выполнении самостоятельного практического задания:

**Задание 2.3. Построение диаграммы декомпозиции в нотации IDEF3**

**Цель работы**: построить диаграмму декомпозиции в нотации IDEF3 одной из работ диаграмм IDEF0, построенных в задании 2.2.

**Методические рекомендации**

IDEF3 - методология моделирования, использующая графическое описание информационных потоков, взаимоотношений между процессами обработки информации и объектов, являющихся частью этих процессов. IDEF3 дает возможность аналитикам описать ситуацию, когда процессы выполняются в определенной последовательности, а также описать объекты, участвующие совместно в одном процессе.

Любая IDEF3-диаграмма может содержать работы, связи, перекрестки и объекты ссылок.

**Работа (Unit of Work, activity).** Изображается прямоугольником с прямыми углами и имеет имя, выраженное отглагольным существительным, обозначающим процесс действия, одиночным или в составе фразы, и номер (идентификатор); другое имя существительное в составе той же фразы обычно отображает основной выход (результат) работы (например, «Изготовление изделия»). Все стороны работы равнозначны. **В каждую работу может входить и выходить ровно по одной стрелке.**



**Связи**. Связи показывают взаимоотношения работ. Все связи в IDEF3 однонаправлены и могут быть направлены куда угодно, но обычно диаграммы IDEF3 стараются построить так, чтобы связи были направлены слева направо. В IDEF3 возможны три вида связей:

|  |  |  |
| --- | --- | --- |
| **Изображение стрелки** | **Название** | **Описание** |
| mhtml:file://W:\Kamskova\МСП\2%20Структурные%20методы\Практика%202.6%20IDEF3.mht!lab05.images/precedence.jpg | Старшая (Precedence) стрелка | сплошная линия, связывающая единицы работ (UOW). Рисуется слева направо или сверху вниз. Показывает, что работа-источник должна закончиться прежде, чем работа-цель начнется |
| mhtml:file://W:\Kamskova\МСП\2%20Структурные%20методы\Практика%202.6%20IDEF3.mht!lab05.images/object_flow.jpg | Потоки объектов (Object Flow) | стрелка с двумя наконечниками, применяется для описания того факта, что объект используется в двух или более единицах работы, например когда объект порождается в одной работе и используется в другой |
| mhtml:file://W:\Kamskova\МСП\2%20Структурные%20методы\Практика%202.6%20IDEF3.mht!lab05.images/relational_link.jpg | Стрелка отношения (Relational Link) | пунктирная линия, использующаяся для изображения связей между единицами работ (UOW), а также между единицами работ и объектами ссылок. Значение задается аналитиком отдельно для каждого случая |

**Перекрестки (Junction).** Окончание одной работы может служить сигналом к началу нескольких работ, или же одна работа для своего запуска может ожидать окончания нескольких работ. Перекрестки используются для отображения логики взаимодействия стрелок при слиянии и разветвлении или для отображения множества событий, которые могут или должны быть завершены перед началом следующей работы. Различают перекрестки для слияния (Fan-in Junction) и разветвления (Fan-out Junction) стрелок. Перекресток не может использоваться одновременно для слияния и для разветвления.

**Типы перекрестков:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Обозначение** | **Наименование** | **Смысл в случае слияния стрелок (Fan-in Junction)** | **Смысл в случае разветвления стрелок (Fan-out Junction)** |
| mhtml:file://W:\Kamskova\МСП\2%20Структурные%20методы\Практика%202.6%20IDEF3.mht!lab05.images/a_and.jpg | Асинхронное «И» (Asynchronous AND) | Все предшествующие процессы должны быть завершены | Все следующие процессы должны быть запущены |
| mhtml:file://W:\Kamskova\МСП\2%20Структурные%20методы\Практика%202.6%20IDEF3.mht!lab05.images/s_and.jpg | Синхронное «И» (Synchronous AND) | Все предшествующие процессы завершены одновременно | Все следующие процессы запускаются одновременно |
| mhtml:file://W:\Kamskova\МСП\2%20Структурные%20методы\Практика%202.6%20IDEF3.mht!lab05.images/a_or.jpg | Асинхронное «ИЛИ» (Asynchronous OR) | Один или несколько предшествующих процессов должны быть завершены | Один или несколько следующих процессов должны быть запущены |
| mhtml:file://W:\Kamskova\МСП\2%20Структурные%20методы\Практика%202.6%20IDEF3.mht!lab05.images/s_or.jpg | Синхронное «ИЛИ» (Synchronous OR) | Один или несколько предшествующих процессов завершены одновременно | Один или несколько следующих процессов запускаются одновременно |
| mhtml:file://W:\Kamskova\МСП\2%20Структурные%20методы\Практика%202.6%20IDEF3.mht!lab05.images/xor.jpg | Исключающее «ИЛИ» XOR (Exclusive OR) | Только один предшествующий процесс завершен | Только один следующий процесс запускается |

**Объект ссылки.** Объект ссылки в IDEF3 выражает некую идею, концепцию или данные, которые нельзя связать со стрелкой, перекрестком или работой. Они используются в модели для привлечения внимания читателя к каким-либо важным аспектам модели. При внесении объектов ссылок помимо имени следует указывать тип объекта ссылки.

mhtml:file://W:\Kamskova\МСП\2%20Структурные%20методы\Практика%202.6%20IDEF3.mht!lab05.images/referent2.jpg

В данной лабораторной работе необходимо одну из работ, находящихся на диаграммах IDEF0 (из задания 2.2), рассмотреть детально с помощью методологии IDEF3. При декомпозиции работы IDEF0 нужно учитывать, что стрелки на диаграммах IDEF0 означают потоки информации или объектов, передаваемых от одной работы к другой.

На диаграммах IDEF3 стрелки могут показывать только последовательность выполнения работ, т.е. они имеют другой смысл, чем стрелки IDEF0. Поэтому при декомпозиции работы IDEF0 в диаграмму IDEF3 стрелки не мигрируют на нижний уровень. Если необходимо показать на дочерней диаграмме IDEF3 те же объекты, что и на родительских диаграммах IDEF0, необходимо использовать объекты ссылки.

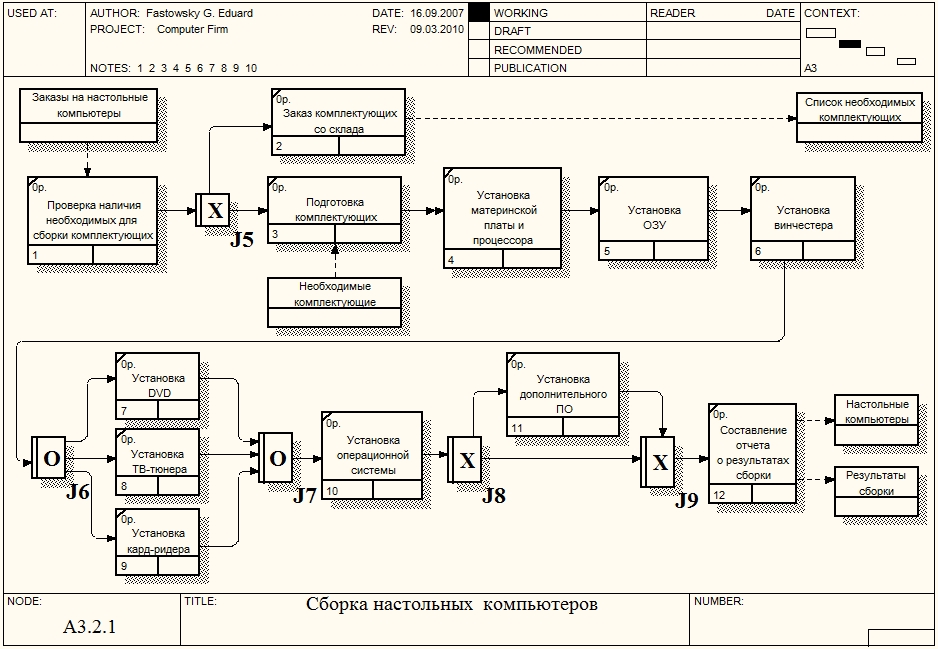
Проведем декомпозицию работы **Сборка настольных компьютеров** диаграммы А3 "Сборка и тестирование компьютеров". Данная работа начинает выполняться, когда поступают заказы на сборку.

1. Первым действием проверяется наличие необходимых для сборки комплектующих и заказ со склада отсутствующих.
2. Далее комплектующие подготавливаются для последующей сборки (освобождение от упаковки, снятие заглушек и т.п.).
3. Следующим шагом начинается непосредственно сам процесс сборки: установка материнской платы в корпус и процессора на материнскую плату, установка ОЗУ и винчестера. Данные действия выполняются всегда, независимо от конфигурации компьютера.
4. Далее по желанию клиента могут быть установлены некоторые дополнительные комплектующие - DVD привод, ТВ-тюнер, кард-ридер. На этом сборка компьютера завершается.
5. Следующим шагом идет установка операционной системы.
6. По желанию клиента также может быть установлено дополнительное программное обеспечение.
7. Последним действием составляется отчет о проделанной работе.

Выделим работу **Сборка настольных компьютеров** диаграммы А3 "Сборка и тестирование компьютеров", нажмем на кнопку "Go to Child Diagram" панели инструментов и выберем нотацию IDEF3.

Дочерние работы всегда можно добавить на диаграмму в процессе ее построения, поэтому число дочерних работ оставим по умолчанию. При создании дочерней диаграммы BPWin переносит граничные стрелки родительской работы, **их необходимо удалить и заменить на объекты ссылок**. Заменим стрелки "Заказы на настольные компьютеры", "Необходимые комплектующие", "Список необходимых комплектующих", "Настольные компьютеры" и "Результаты сборки" на объекты ссылок - кнопка "**Referent**" на панели инструментов, в появившемся окне выбрать переключатель "**Arrow**" и выбрать из списка нужное название

Далее начинаем располагать на диаграмме работы, отражающие указанные выше действия, выполняемые при сборке компьютеров. Итоговая диаграмма декомпозиции работы в нотации IDEF3 имеет вид:



Рассмотрим основные особенности этой диаграммы. После проверки наличия необходимых для сборки комплектующих возможно одно из двух действий - или заказ со склада недостающих комплектующих, или, если все комплектующие в наличии, их подготовка. Поэтому мы поставили перекресток разветвления типа **"Исключающее ИЛИ".**

Работы "Подготовка комплектующих" и "Установка материнской платы и процессора" соединены связью "Поток объектов". Тем самым мы показываем, что между этими работами передаются объекты. Все последующие работы соединяются связями "старшая стрелка", поскольку они только показывают последовательность действий над одними и теме же объектами.

После установки винчестера возможна установка DVD привода, ТВ-тюнера, кард-ридера или любая их комбинация. Поэтому мы поставили перекресток разветвления типа "**Асинхронное ИЛИ".** Такой же перекресток стоит и после завершения этих работ.

Далее после установки операционной системы может быть установлено дополнительное ПО, или же сразу формируется отчет, поэтому мы поставили перекресток разветвления типа **"Исключающее ИЛИ".** За перекрестком разветвления типа "Исключающее ИЛИ" может следовать только такой же перекресток слияния, поэтому перед работой "Составление отчета о результатах сборки" мы поставили такой же.

**Задание 2.4. Построение диаграммы декомпозиции в нотации DFD**

**Цель работы:** построить диаграмму декомпозиции в нотации DFD одной из работ диаграмм IDEF0, построенных в задании 2.2.

**Методические рекомендации**

Диаграммы потоков данных (Data flow diagram, DFD) используются для описания документооборота и обработки информации. Подобно IDEF0, DFD представляет моделируемую систему как сеть связанных между собой работ. Их можно использовать как дополнение к модели IDEF0 для более наглядного отображения текущих операций документооборота в корпоративных системах обработки информации.

Главная цель DFD - показать, как каждая работа преобразует свои входные данные в выходные, а также выявить отношения между этими работами.

Любая DFD-диаграмма может содержать работы, внешние сущности, стрелки (потоки данных) и хранилища данных.

**Работы.** Работы изображаются прямоугольниками с закругленными углами, смысл их совпадает со смыслом работ IDEF0 и IDEF3. Так же как работы IDEF3, они имеют входы и выходы, но не поддерживают управления и механизмы, как IDEF0. Все стороны работы равнозначны. В каждую работу может входить и выходить по несколько стрелок.

**Внешние сущности.** Внешние сущности изображают входы в систему и/или выходы из нее. Одна внешняя сущность может одновременно предоставлять входы (функционируя как поставщик) и принимать выходы (функционируя как получатель). **Внешняя сущность представляет собой материальный объект, например заказчики, персонал, поставщики, клиенты, склад. Определение некоторого объекта или системы в качестве внешней сущности указывает на то, что они находятся за пределами границ анализируемой системы.** Внешние сущности изображаются в виде прямоугольника с тенью и обычно располагаются по краям диаграммы

**Стрелки (потоки данных).**Стрелки описывают движение объектов из одной части системы в другую (отсюда следует, что диаграмма DFD не может иметь граничных стрелок). Поскольку все стороны работы в DFD равнозначны, стрелки могут могут начинаться и заканчиваться на любой стороне прямоугольника. Стрелки могут быть двунаправлены.

**Хранилище данных.** В отличие от стрелок, описывающих объекты в движении, хранилища данных изображают объекты в покое. **Хранилище данных - это абстрактное устройство для хранения информации, которую можно в любой момент поместить в накопитель и через некоторое время извлечь, причем способы помещения и извлечения могут быть любыми. Оно в общем случае является прообразом будущей базы данных, и описание хранящихся в нем данных должно соответствовать информационной модели (Entity-Relationship Diagram).**

**Декомпозиция работы IDEF0 в диаграмму DFD.** При декомпозиции работы IDEF0 в DFD необходимо выполнить следующие действия:

* удалить все граничные стрелки на диаграмме DFD;
* создать соответствующие внешние сущности и хранилища данных;
* создать внутренние стрелки, начинающиеся с внешних сущностей вместо граничных стрелок;
* стрелки на диаграмме IDEF0 затоннелировать

Строго придерживаться правил нотации DFD не всегда удобно, поэтому BPWin позволяет создавать в DFD диаграммах граничные стрелки.

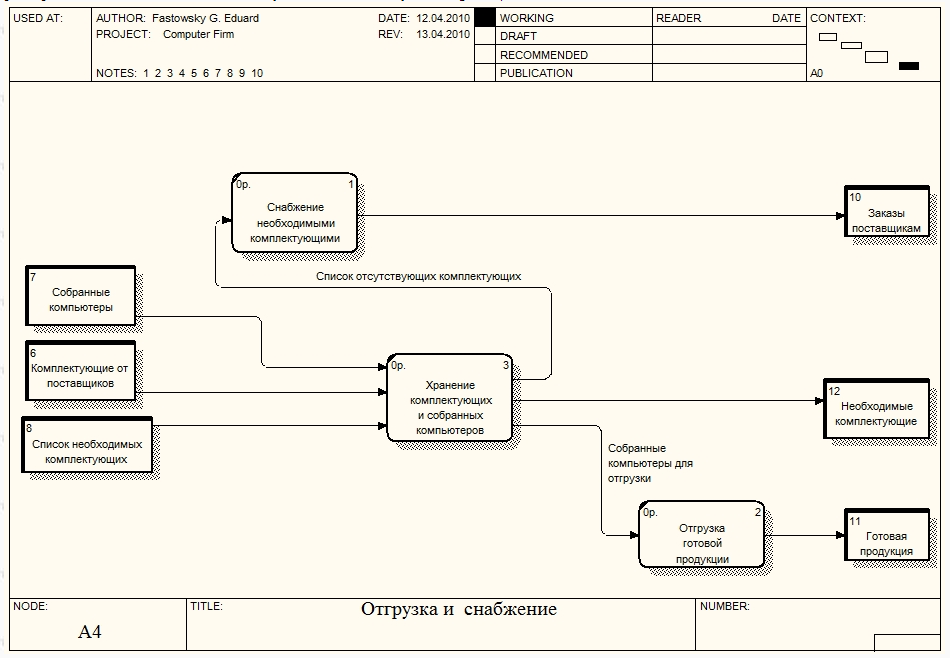
**Построение диаграммы декомпозиции.** Проведем декомпозицию работы *Отгрузка и получение* диаграммы А0 "Деятельность предприятия по сборке и продаже компьютеров и ноутбуков". В этой работе мы выделили следующие дочерние работы:

* снабжение необходимыми комплектующими - занимается действиями, связанными с поиском подходящих поставщиков и заказом у них необходимых комплектующих
* хранение комплектующих и собранных компьютеров
* отгрузка готовой продукции - все действия, связанные с упаковкой, оформлением документации и собственно отгрузкой готовой продукции

Выделим работу *Отгрузка и получение* диаграммы А0 "Деятельность предприятия по сборке и продаже компьютеров и ноутбуков", нажмем на кнопку "Go to Child Diagram" панели инструментов и выберем нотацию DFD.

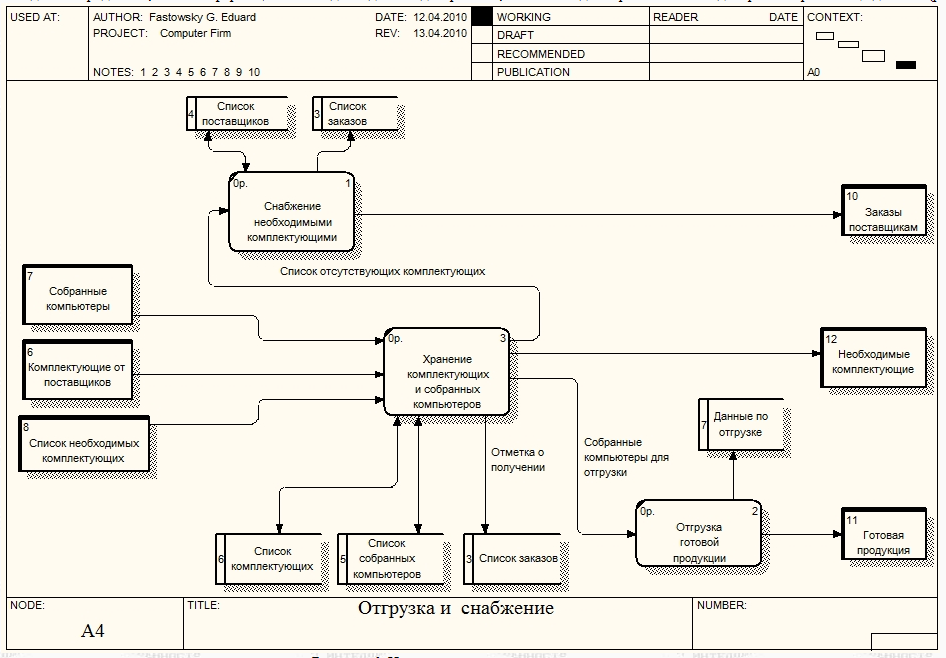
При создании дочерней диаграммы BPWin переносит граничные стрелки родительской работы, их необходимо удалить и заменить на внешние сущности. Стрелки механизмов, стрелки управления "Правила и процедуры", "Управляющая информация" и стрелку выхода "Отчеты" на дочерней диаграмме задействованы не будут, чтоб не загромождать диаграмму менее существенными деталями. Остальные стрелки заменим на внешние сущности - кнопка "External Reference Tool" на панели инструментов, в появившемся окне выбрать переключатель "Arrow" и выбрать из списка нужное название:

Далее разместим дочерние работы, свяжем их со внешнеми сущностями и между собой:



Центральной здесь является работа "Хранение комплектующих и собранных компьютеров". На ее вход поступают собранные компьютеры и полученные от поставщиков комплектующие, а также список необходимых для сборки компьютеров комплектующих. Выходом этой работы будут необходимые комплектующие (если они есть в наличии), список отсутствующих комплектующих, передаваемый на вход работы "Снабжение необходимыми комплектующими" и собранные компьютеры, передаваемые на отгрузку. Выходами работ "Снабжение необходимыми комплектующими" и "Отгрузка готовой продукции" будут, соответственно, заказы поставщикам и готовая продукция.

Следующим шагом необходимо определить, **какая информация необходима для каждой работы**, т.е. необходимо разместить на диаграмме хранилища данных.



Работа "Снабжение необходимыми комплектующими" работает с информацией о поставщиках и с информацией о заказах, сделанных у этих поставщиков. Стрелка, соединяющая работу и хранилище данных "Список поставщиков" двунаправленная, т.к. работа может как получать информацию о имеющихся поставщиках, так и вносить данные о новых поставщиках. Стрелка, соединяющая работу с хранилищем данных "Список заказов" однонаправленная, т.к. работа только вносит информацию о сделанных заказах.

Работа "Хранение комплектующих и собранных компьютеров" работает с информацией о получаемых и выдаваемых комплектующих и собранных компьютеров, поэтому стрелки, соединяющая работу с хранилищами данных "Список комплектующих" и "Список собранных компьютеров" двунаправленные. Также эта работа при получении комплектующих должна делать отметку о том, что заказ поставщикам выполнен. Для этого она связана с хранилищем данных "Список заказов" однонаправленной стрелкой. Обратите внимание, что на DFD диаграммах одно и тоже хранилище данных может дублироваться.

Наконец, работа "Отгрузка готовой продукции" должна хранить информацию по выполненным отгрузкам. Для этого вводится соответствующее хранилище данных - "Данные по отгрузке".

## Раздел 3. Метод Сущность-связь (ER)

**Тема 3.1. Метод Сущность-связь (ER)**

**Задание 3.1. Построение ER-диаграммы средствами Microsoft Visio**

1. Используя Microsoft Visio спроектировать ER-диаграмму для предложенной предметной области.
2. Для построения диаграммы используйте нотацию Block Diagram из раздела Block Diagram и соответствующие блоки.

**Предметная область**

В музее, в котором постоянно проходят выставки, экскурсии и прочие мероприятия существует несколько залов. Каждый зал характеризуется некоторым именем и площадью.

В каждом зале в определенный период года проводятся выставки. Каждая выставка может размещаться только в одном зале.

На выставку выставляются экспонаты музея. Экспонаты музея относятся к определенной категории (например, живопись, скульптура и пр.) и имеют состояние (на реставрации, в зале, в хранилище и пр.). Экспонаты также имеют следующие характеристики – наименование и автор.

В каждом зале может сотрудник музея может проводить экскурсию, которая проводится строго по графику (в определенный день недели, в определенное время и имеет стоимость). Каждую экскурсию в конкретном зале проводит тот или иной сотрудник.

Хранятся сведения о сотруднике: его ФИО и должность.

**Методические рекомендации**

**1 шаг Выделяем сущности**

Для описания объектов предметной области музея выделим следующие сущности:

**Залы**

**Выставки**

**Экспонаты музея**

**Экскурсии**

**График**

**Сотрудники**

**2 шаг Определяем идентификаторы сущностей (ключи)**

Определяем идентификаторы для каждой сущности

**Залы -** искусственный идентификатор – **код\_зала**.

**Выставки–** искусственный идентификатор – **код\_выставки**

**Экспонаты музея –** искусственный идентификатор – **код\_экспоната**

**Экскурсии–** искусственный идентификатор– **код\_экскурсии**

**График** - искусственный идентификатор– **код\_графика**

**Сотрудники-** искусственный идентификатор – **код\_сотрудника**

**3 шаг Определяем атрибуты сущностей**

Определяем остальные атрибуты для каждой сущности

**Залы – Наименование, площадь**

**Выставки – Дата Начала, Дата окончания**

**Экспонаты музея – Наименование, Автор, Категория, Состояние**

**Экскурсии – Наименование, продолжительность,**

**График – День недели, Время, Стоимость**

**Сотрудники – ФИО, Должность**

**4 шаг Формируем связи**

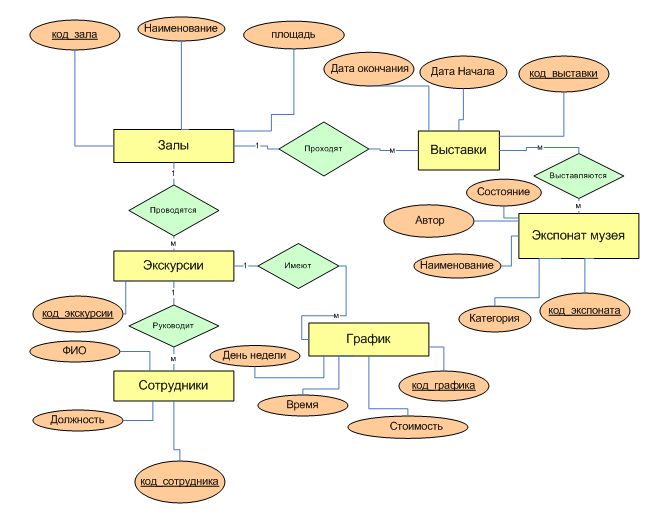
1. В зале **Проходят** выставки (в одном зале может проходить много выставок, но одна выставка проходит только в одном зале (по условию задачи)). Связь 1:М.
2. В зале **Проводятся** экскурсии. Связь 1:М.
3. Экскурсии **Имеют** график проведения. (одна экскурсия имеет один конкретный график, но по одному графику проводится несколько экскурсий (например, в разных залах). Связь 1:М.
4. Каждой экскурсией **Руководит** сотрудник. Связь 1:М.
5. На каждой выставке В**ыставляются** определенные экспонаты. На одной выставке может быть выставлено много экспонатов. Каждый экспонат может участвовать на нескольких выставках. Связь М:М.

**Задание 3.2. Проектирование схемы реляционной БД средствами Visio**

На основании разработанной в Задании 3.1 ER-диаграммы разработать схему реляционной БД.

**Методические рекомендации**

ER-диаграмма имеет примерно следующий вид:



Сущности **Залы, Выставки, Экскурсии, Экспонаты, Сотрудники и График** преобразуются в соответствующие таблицы.

1. Проектируем таблицу **Залы,** котораяимеет искусственный ключ **Код\_зала** и атрибуты **Наименование зала и Площадь**. Создайте эту таблицу и спроектируйте атрибуты.
2. Проектируем таблицу **График,** котораяимеет искусственный ключ **Код\_графика** и атрибуты **День недели, Время и Стоимость**. Создайте эту таблицу и спроектируйте атрибуты.
3. Проектируем таблицу **Экспонаты,** котораяимеет ключ **Код\_Экспоната,** дваатрибута **Автор и Наименование**. Еще два атрибута **Категория** и **Состояние** преобразуются в отдельные таблицы:
4. **Категория** – с ключом **Код**\_**категории** и с атрибутами **Категория** и **Описание**
5. **Состояние** - с ключом **Код**\_**состояния** и с атрибутом **Состояние**
6. Для организации связи между таблицами **Экспонаты** и **Состояние** необходимо в таблице **Экспонаты** добавить внешний ключ **Код\_Состояния и** спроектировать связь 1:М.
7. Аналогичным образом проектируем связь между таблицами **Экспонаты** и **Категория.**
8. Проектируем таблицу **Экскурсии с** ключом Код\_экскурсии. Для организации связи между таблицами **Экскурсии** и **График** необходимо в таблице **Экскурсии** добавить внешний ключ **Код\_Графика и** спроектировать связь 1:М.
9. Для организации связи между таблицами **Экскурсии** и **Сотрудники** необходимо в таблице **Экспонаты** добавить внешний ключ **Код\_Состояния и** спроектировать связь 1:М.
10. Проектируем таблицу **Сотрудники,** котораяимеет ключ **Код\_сотрудника** иатрибут **ФИО**. Атрибут **Должность** преобразуются в отдельную таблицу **Должности** с ключом **Код\_Должности** и атрибутом **Название**.
11. Для организации связи между таблицами **Сотрудники** и **Должность** необходимо в таблице **Сотрудники** добавить внешний ключ **Код\_Должности и** спроектировать связь 1:М.
12. Для организации связи между таблицами **Экскурсии** и **Залы** необходимо в таблице **Экскурсии** добавить внешний ключ **Код\_Зала и** спроектировать связь 1:М.
13. Для организации связи между таблицами **Выставки** и **Залы** необходимо в таблице **Выставки** добавить внешний ключ **Код\_Зала и** спроектировать связь 1:М.
14. . В ER-диаграмме присутствует связь М:М между сущностями **Выставки** и **Экспонаты**. Связь данного типа преобразуется в отдельную таблицу **Экспонат\_выставки**, которая имеет ключ **Код\_Экспоната\_Выставки**.
15. Для организации связи между таблицами **Экспонат\_выставки** и **Экспонат** необходимо в таблице **Экспонат\_выставки** добавить внешний ключ **Код\_Экспоната и** спроектировать связь 1:М.
16. Для организации связи между таблицами **Экспонат\_выставки** и **Выставка** необходимо в таблице **Экспонат\_выставки** добавить внешний ключ **Код\_Выставки и** спроектировать связь 1:М.
17. Схема реляционной базы данных спроектирована. Она имеет примерный вид, представленный на рисунке ниже.

**Задание 3.3. Проектирование схемы реляционной БД средствами StarUML**

Средствами StarUML спроектировать схему реляционной базы данных, созданную в задании 3.2.

**Методические рекомендации**

StarUML является программной платформы моделирования, которая поддерживает UML (Unified Modeling Language). Он основан на UML версии 1.4 и предоставляет одиннадцать различных типов диаграмм.

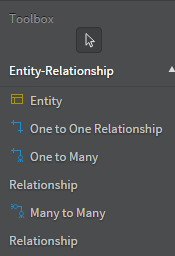
В StarUML основной структурой единицей является проект. В проекте содержится модель системы, состоящая из диаграмм. Проектные файлы сохраняются в формате XML и имеют расширение .UML

Чтобы начать разработку программного обеспечения, нужно инициировать новый проект. Выберите меню File -> New From Templatе – Data Model.

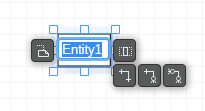
Внешний вид экрана StarUML: в центре рабочее поле для построения диаграмм. Справа наверху – браузер модели, справа внизу – окно редактирования. Слева внизу – панель инструментов. Слева наверху список рабочих диаграмм.

Внизу Diagram Thumbnails – место хранения пиктограмм диаграмм.

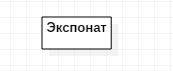
1. На панели инструментов представлены пиктограммы сущностей и трех видов связи



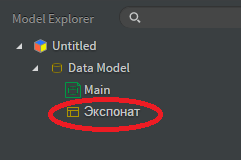
1. Для размещения на диаграмме отношения **Экспонаты** следует на панели инструментов выбрать пиктограмму **Entity** и щелкнуть по рабочему полю диаграммы. Получится



1. Введите имя отношения **Экспонаты**.



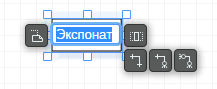
1. Обратите внимание, как изменился браузер модели. В браузере появилась наша диаграмма (Main) и отношение **Экспонаты**



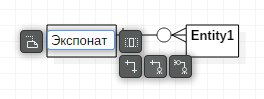
1. Чтобы разместить на диаграмме другие отношения и становить связи можно использовать два способа:

**1 способ**

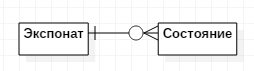
* 1. можно воспользоваться кнопками горячего синтаксиса. Для этого дважды щелкните по отношению **Экспонаты**. Вокруг отношения появятся кнопки горячего синтаксиса, которые позволят для данного отношения добавить атрибут, один из видов связи или заметку.



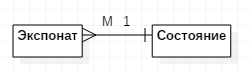
* 1. Нажмите кнопку Add One to Many (добавить связь один ко многим). Появится новое отношение, связанное с отношением **Экспонаты**



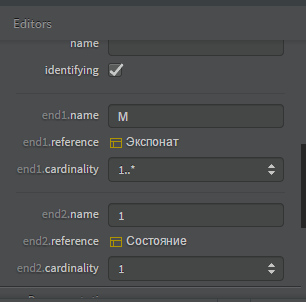
* 1. Назовите его **Состояние**



* 1. В данном случае необходимо изменить направленность связи и обозначить ее тип



* 1. Для этого необходимо выделить связь, в окне редактирования установить следующие свойства.



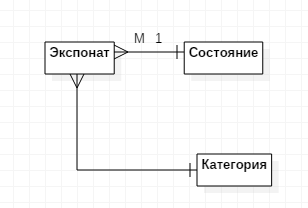
* 1. В поле Name при необходимости можно внести имя связи.

**2 способ**

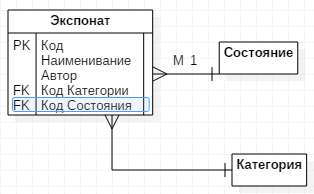
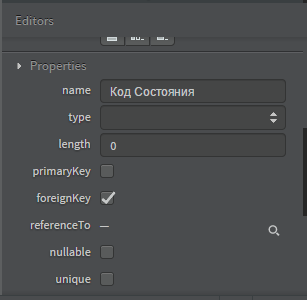
1. Добавьте на диаграмму новое отношение **Категория**
2. Щелкните по пиктограмме One to Many Relationship, а потом по отношению **Категория.** Получится**:**



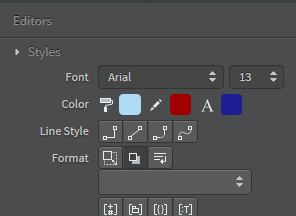
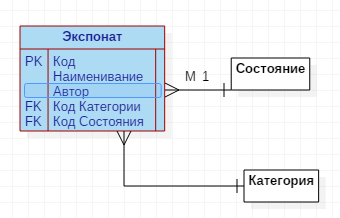
1. Переместите и прикрепите нужный конец связи к отношению **Экспонат**. При необходимости установить тип связи.



1. Добавьте к отношению **Экспонат** атрибуты, воспользовавшись кнопкой горячего синтаксиса **Add Column** или воспользовавшись комбинацией клавиш **Ctrl+Enter**.
2. Установите первичные и внешние ключи. Для этого выделите необходимый атрибут и в окне редактирования свойств атрибута поставьте соответствующий флаг.



1. При необходимости поменяйте цвет фона, шрифта и линий сущности в окне редактирования свойств.



1. Постройте схему согласно рисунку.



## Раздел 4. Визуальное моделирование на языке UML

**Тема 4.1. Визуальное моделирование на языке UML**

**Задание 4.1 Построение диаграммы вариантов использования средствамиStarUml**

Необходимо построить диаграмму вариантов использования для следующей предметной области:

**Описание предметной области**

Фирма имеет собственный парк автомобилей. Автомобиль имеет следующие характеристики: регистрационный номер, марка, модель, мощность, цвет, стоимость проката в сутки.

Новые автомобили поступают на фирму по документу "Поступление автомобиля", в котором указываются все сведения автомобиля, дата поступления, цена поступления, процент износа, государственный регистрационный номер и дата регистрации.

Старые (поврежденные) автомобили списываются с баланса фирмы документом "Акт о списании", в котором указывается дата списания, автомобиль, причины списания.

Каждый автомобиль может быть предоставлен в аренду клиенту. Для проката клиент предоставляет данные паспорта (ФИО, дата рождения, адрес, номер паспорта, когда и кем выдан паспорт), при этом формируется база данных клиента.

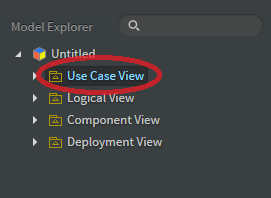
При формировании документа аренды указывается: дата и номер документа, сведения о клиенте, сведения об автомобиле, дата начала аренды, срок и стоимость аренды.

Расчетный час - 7 часов утра.

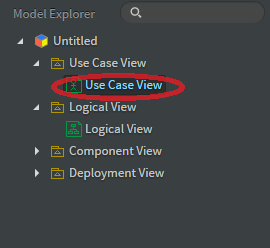
**Методические рекомендации**

Создайте новый проект. При создании нового проекта изначально активным становится логическое построение модели, здесь можно построить диаграмму классов.

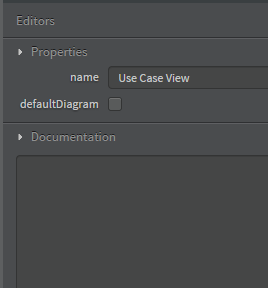
Для построения диаграммы ВИ необходимо в браузере модели дважды щелкнуть по представлению Use Case View.



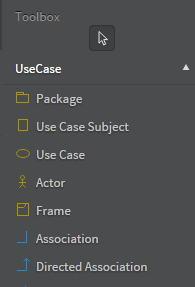
Станет доступным Диаграмма вариантов использования.



Выбираем Двойным щелчком диаграмму Use Case View. При этом станет активно соответствующее окно документации



И панель инструментов



В диаграмме прецедентов доступны следующие элементы.

• Актер

• Прецедент

• Ассоциация

• Направленная ассоциация

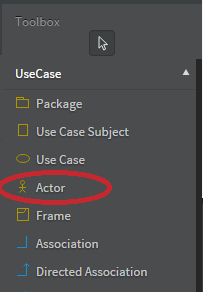
Остальные элементы пока рассматривать не будем.

**Актер**

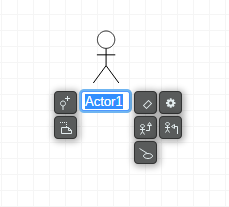
Актер определяет множество ролей, которые пользователи некоторой сущности могут играть при взаимодействии с этой сущностью. Актер может играть отдельную роль в отношении с любым прецедентом, с которым он связан.

**Процедура создания актера**

Чтобы создать актера, нажмите кнопку [Toolbox] -> [UseCase] -> [Actor]



и щёлкните позицию на диаграмме, куда Вы хотите поместить актера. Актер изображается в виде человечка или прямоугольника со значком человечка, если используется декоративный режим отображения.

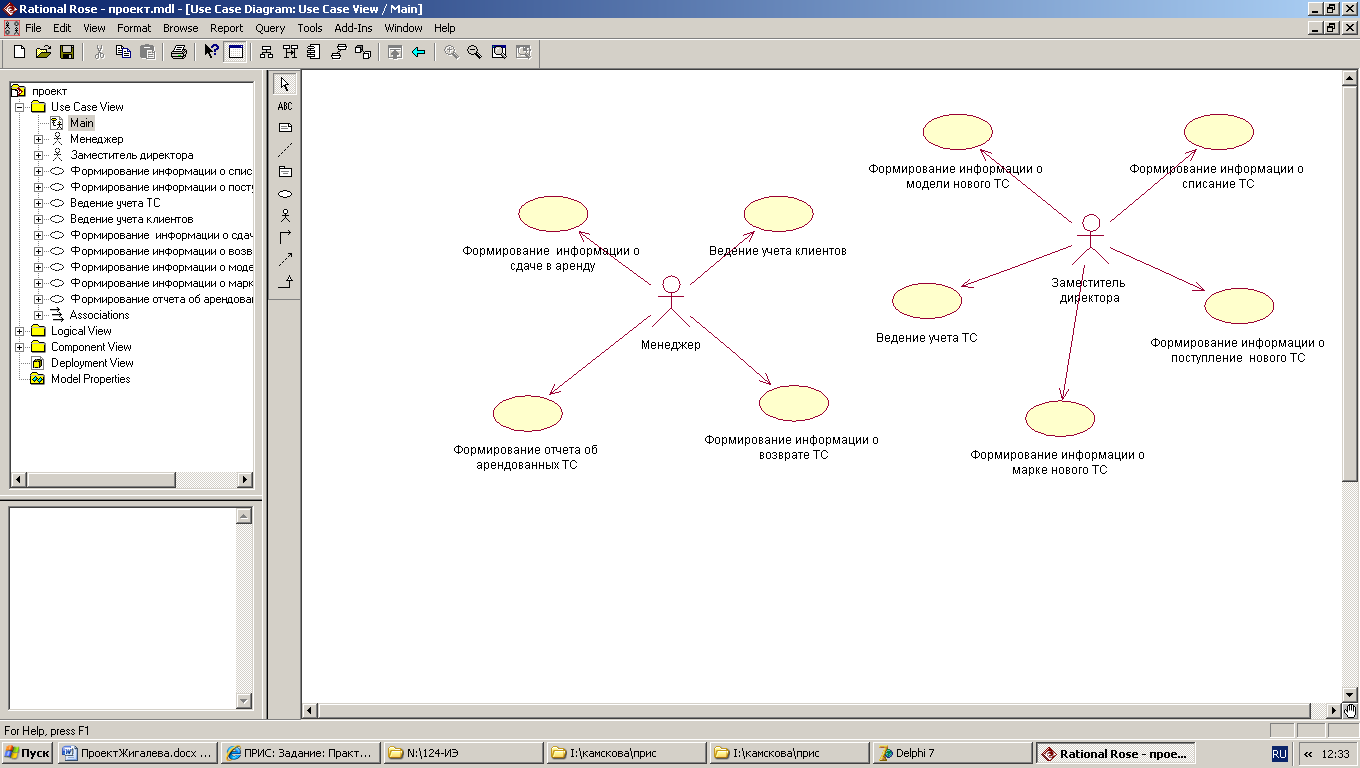


При этом вокруг элемента отображаются кнопки «горячего диалога».

Вместо названия по умолчанию Actor1 введите название Менеджер.

Чтобы отображать актера в декоративном режиме, выберите пункт меню [Format] -> [Stereotype Display] -> [Decoration]. Посмотрите, как будет выглядеть актер в этом режиме и вернитесь назад в активный режим. Внешний вид элемента можно изменить при помощи команды Format-Stereotype Display в контекстном меню.

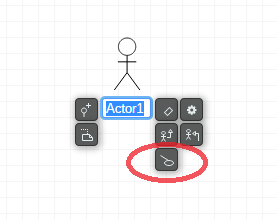
Диаграмма вариантов использования для нашей системы будет выглядеть следующим образом:



Актер Менеджер инициирует четыре варианта использования.

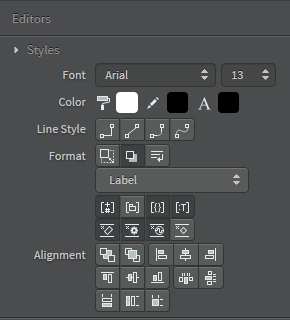
**Процедура создания нескольких прецедентов для одного актера**

Чтобы одновременно создать несколько прецедентов, связанных с одним актером, используйте кнопки горячего диалога, который вызывается двойным щелчком по соответствующему элементу диаграммы. Для добавления четырех вариантов использования четыре раза щелкните по кнопке Add Associated Use Case.

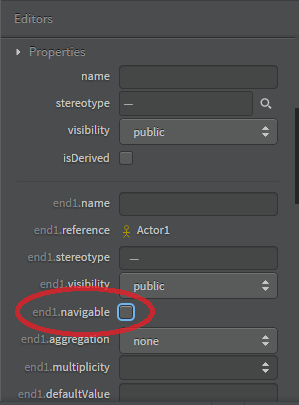


Добавьте соответствующие названия прецедентов.

Можно изменить внешний вид как самого элемента диаграммы ВИ, так и шрифта. Для этого выделите соответствующий элемент и в окне редактирования в разделе Styles выберите соответствующие форматы элемента.



Обратите внимание, что спроектированные связи между актером и вариантами использования не имеют навигацию. Чтобы задать навигацию необходимо выделить соответствующую ассоциацию и в окне редактирования в разделе Properties снимите флаг у свойства end1.navigable.



**Прецедент**

Конструкция прецедента используется, чтобы определить поведение системы или другой семантической сущности, не показывая внутреннюю структуру этой сущности. Каждый прецедент определяет последовательность действий, включая варианты, которые сущность может выполнить, взаимодействуя с актерами.

**Процедура создания прецедента**

Нажмите кнопку [Toolbox] -> [UseCase] и щёлкните позицию на диаграмме, где нужно разместить прецедент.

Прецедент может отображаться в текстовом, декоративном или графическом формате. Чтобы изменить формат отображения прецедента, выберите пункт меню [Format] -> [Stereotype Display] или соответствующую команду контекстного меню.

Создайте таким образом оставшиеся пять прецедентов.

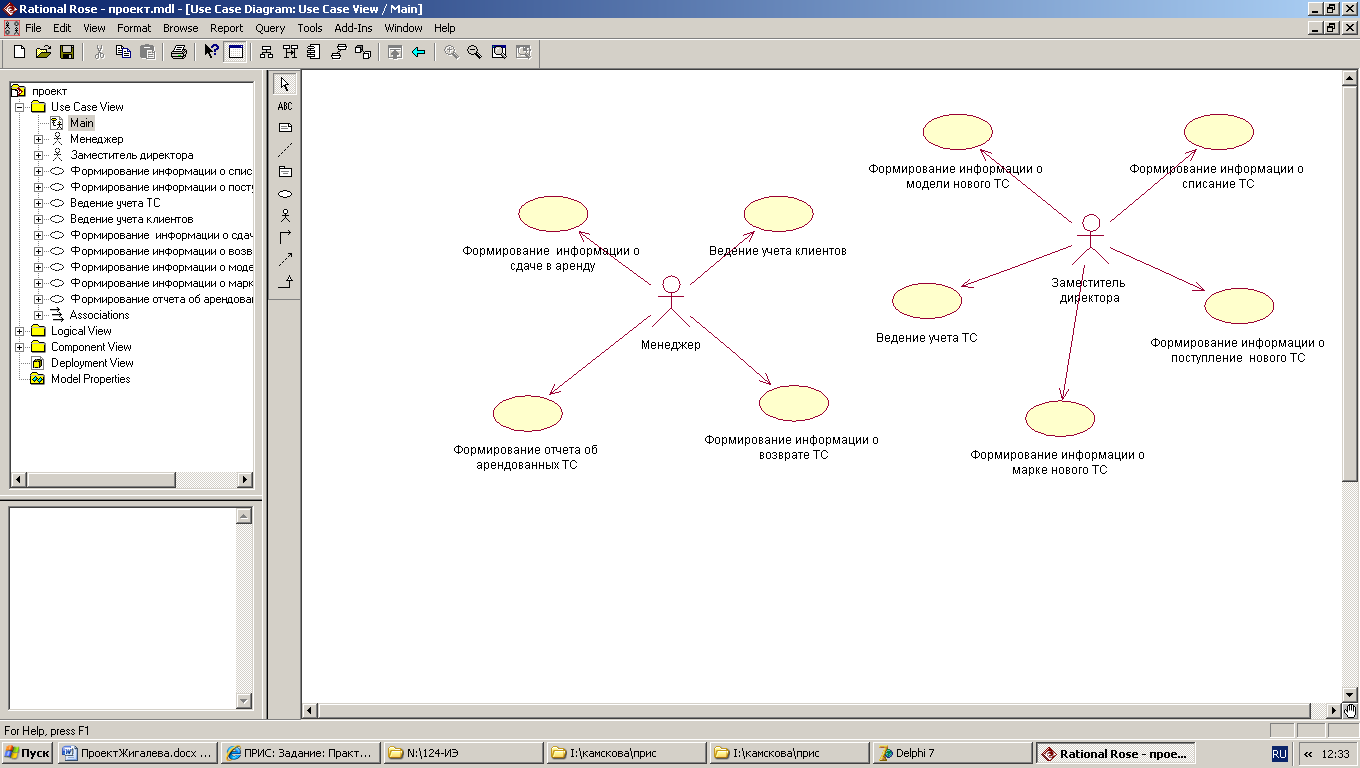
Создайте актера Заместитель директора.

Осталось добавить ассоциации от актера к прецедентам.

**Чтобы создать направленную ассоциацию**

Чтобы создать ассоциацию, нажмите кнопку [Toolbox] -> [UseCase] -> [DirectAssociation], проведите линию от исходного элемента к целевому элементу. Проводить линию нужно в направлении стрелки. Добавьте соответствующие ассоциации на диаграмму.

Полученная диаграмма должна иметь примерно такой вид

Внимательно ознакомившись с описанием предметной области, добавьте недостающего актера и вариант использования на диаграмму.

**Задание 4.2. Проектирование диаграммы классов и диаграммы последовательности средствами StarUML**

**Описание предметной области**

Фирма имеет собственный парк автомобилей. Автомобиль имеет следующие характеристики: регистрационный номер, марка, модель, мощность, цвет, стоимость проката в сутки.

Новые автомобили поступают на фирму по документу "Поступление автомобиля", в котором указываются все сведения автомобиля, дата поступления, цена поступления, процент износа, государственный регистрационный номер и дата регистрации.

Старые (поврежденные) автомобили списываются с баланса фирмы документом "Акт о списании", в котором указывается дата списания, автомобиль, причины списания.

Каждый автомобиль может быть предоставлен в аренду клиенту. Для проката клиент предоставляет данные паспорта (ФИО, дата рождения, адрес, номер паспорта, когда и кем выдан паспорт), при этом формируется база данных клиента.

При формировании документа аренды указывается: дата и номер документа, сведения о клиенте, сведения об автомобиле, дата начала аренды, срок и стоимость аренды.

Расчетный час - 7 часов утра.

**Методические рекомендации**

**Построение диаграммы классов для варианта использования «Выдача автомобиля в аренду»**

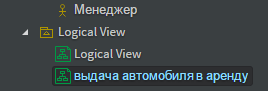
Построим диаграмму классов для варианта использования «Выдача автомобиля в аренду». Определим следующие классы для данного варианта использования:

Граничные классы: Форма справочника автомобилей, Форма справочника клиентов, Форма аренды автомобиля.

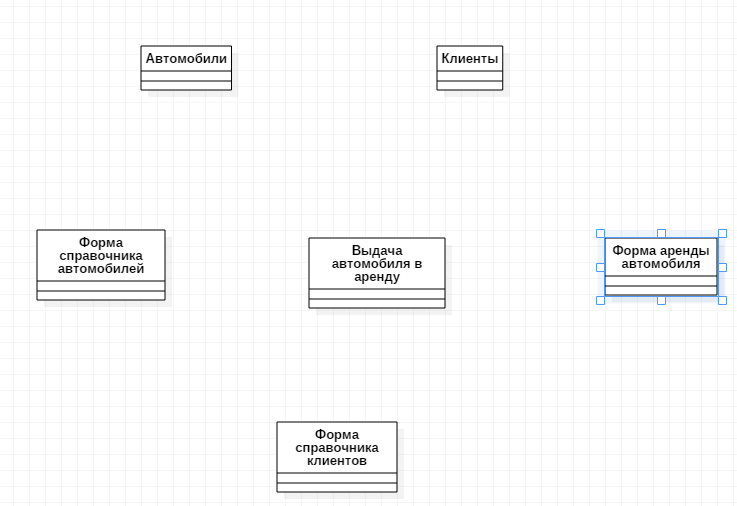
Классы-сущности: Автомобили, Клиенты, Аренда автомобилей.

**Выполнение задания**

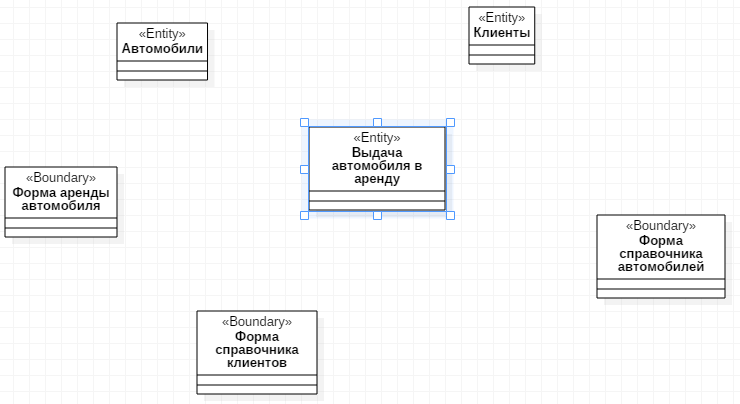
1. Откройте файл с диаграммой вариантов использования, созданной в SratUML.
2. В логическом браузере выбрать раздел Logical View и вызвать контекстное меню.
3. Выбрать команду Add Diagram – Class Diagram. В окне редактирования ввести название диаграммы Выдача автомобиля в аренду. Браузер должен выглядеть примерно так:



1. Разместите на рабочем поле шесть выделенных классов и присвойте им имена как на рисунке.



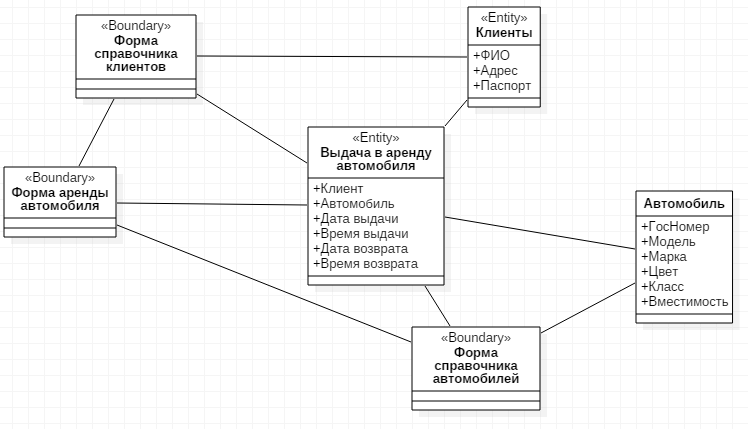
1. Задайте соответствующие стереотипы классам. Для этого в разделе описания свойств в поле Stereotype введите стереотипы Entity и Boundary.



1. Расставьте ассоциации классов как показано на рисунке



1. Для классов-сущностей введите атрибуты

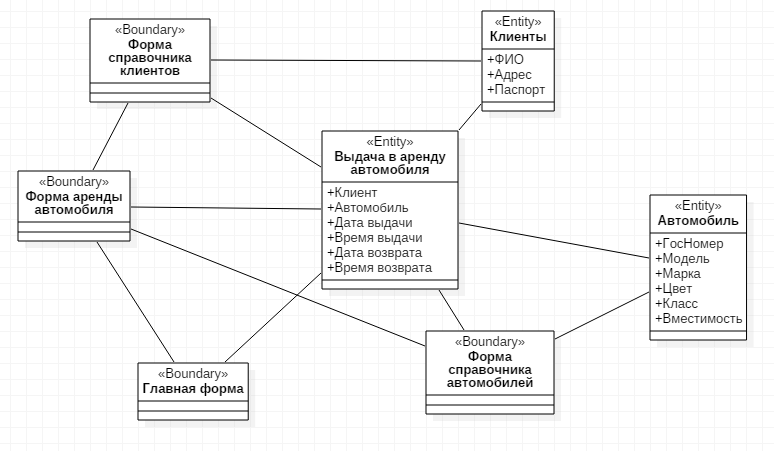


**Построение диаграммы последовательности для варианта использования «Выдача автомобиля в аренду»**

Проанализируем исполнение варианта использования «Выдача автомобиля в аренду» (сценарий варианта использования).

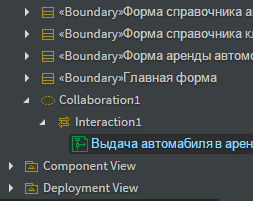
1. На главной форме менеджер выбирает команду создания нового документа аренды
2. Открывается Форма аренды автомобиля
3. В нее менеджер заносит информация для аренды: Дату аренды, срок.
4. Для ввода информации о клиенте менеджер открывает справочник клиентов.
5. Открывается форма справочника клиентов
6. В нее подгружается информация из БД клиентов.
7. Менеджер выбирает необходимого клиента.
8. Информация о выбранном клиенте появляется в форме аренды автомобиля.
9. Форма справочника клиента закрывается
10. Для ввода информации об арендуемом автомобиле менеджер открывает справочник автомобилей.
11. Открывается форма справочника автомобилей
12. В нее подгружается информация из БД автомобилей.
13. Менеджер выбирает необходимый автомобиль.
14. Информация о выбранном авто появляется в форме аренды автомобиля.
15. Форма справочника автомобилей закрывается
16. Менеджер выбирает команду сохранить на Форме аренды автомобиля
17. Данные из формы сохраняются в соответствующей таблице БД,
18. Форма аренды автомобиля закрывается.
19. Открывается форма журнала документов «Аренда»,
20. В нее подгружаются сведения о документах аренды, в том числе и о только что созданном документе.

Как видно из сценария, появилось еще два граничных класса; Главная форма и Форма журнала аренды. Если предположить, что главная форма в данной информационной системе может одновременно быть и формой журнала документов аренды (что естественно для небольших информационных систем), то добавьте на диаграмму классов еще один класс «Главная форма»

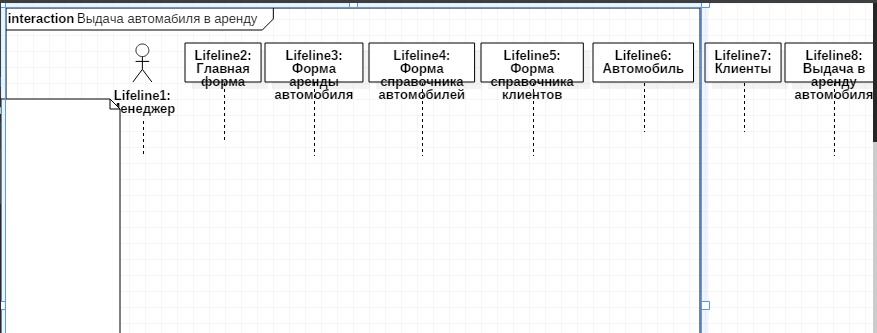


Построим диаграмму последовательности

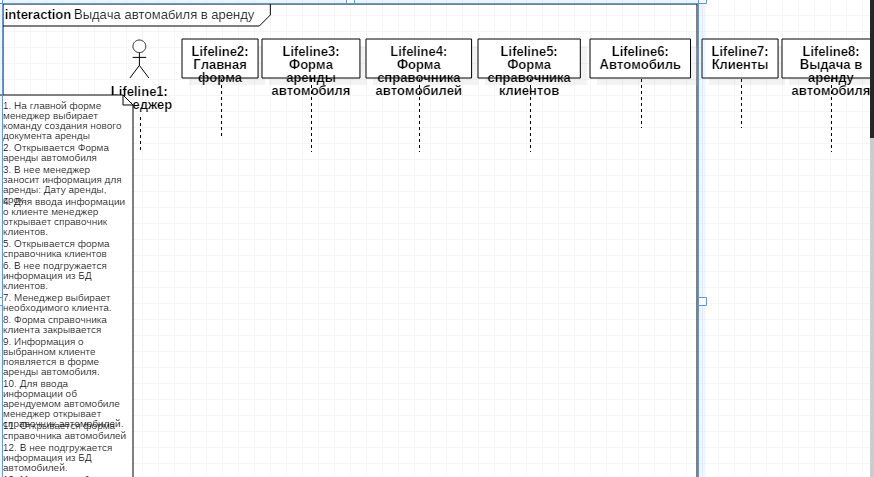
1. Вызовите контекстное меню на разделе Logical View и создайте диаграмму последовательности Sequence Diagram. Назовите ее Выдача автомобиля в аренду



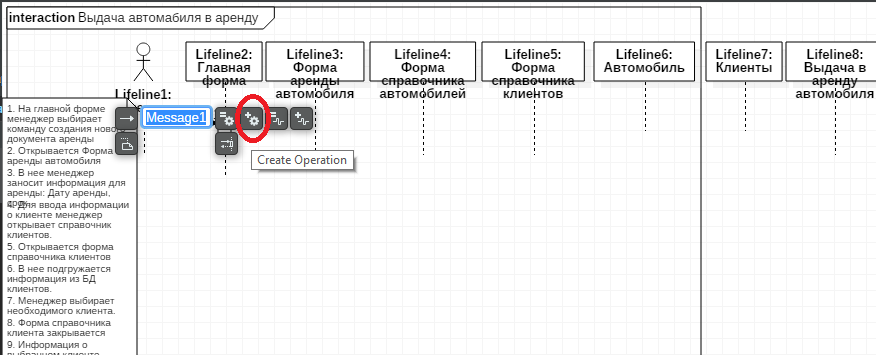
1. Перетащите из панели инструментов объект Note, а из браузера актера и спроектированные ранее классы на рабочее поле диаграммы последовательности, как показано на рисунке



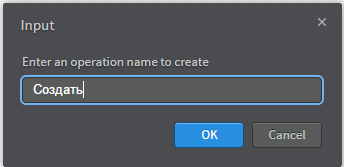
1. Поместите в поле Note сценарий варианта использования.



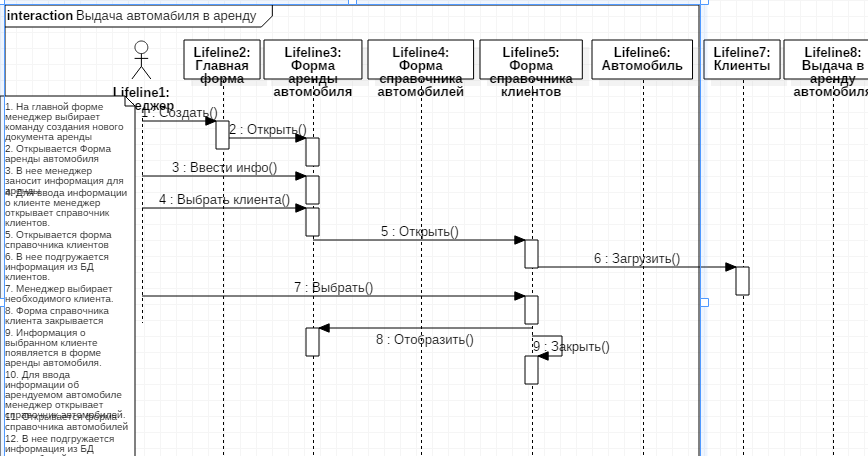
1. Будем проектировать операции классов, соответствующие описанию в сценарии. Первое описание: На главной форме менеджер выбирает команду создания нового документа аренды. Создаем сообщение (Massage) от линии жизни актера к классу Главная форма и сразу же соотносим ее с операцией. Для этого:
2. Нажмите кнопку горячего синтаксиса Create operation



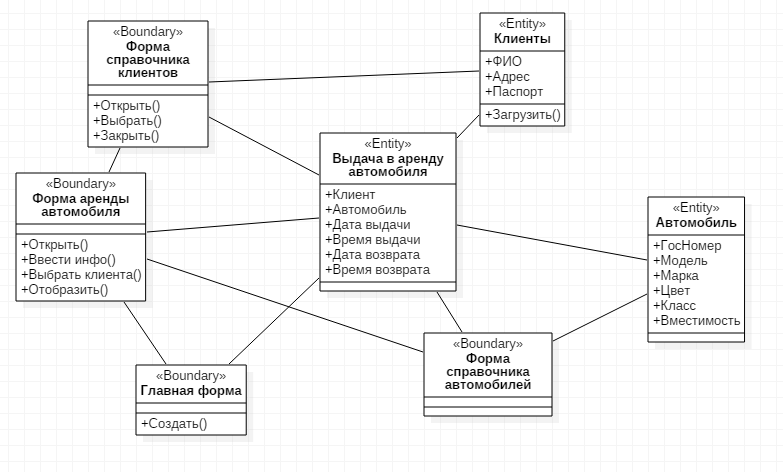
1. И введите название операции **Создать**.



1. Создайте таким образом все операции данного варианта использования. Первые девять операций выглядят примерно так:



1. Следите за тем, чтобы сообщения создавались напротив соответствующего описания в поле Note.
2. Обратите внимание, что в браузере отображаются созданные операции у соответствующих классов. Соответственно отображаются созданные операции на диаграмме классов. Диаграмма классов с первыми девятью операциями выглядит примерно так:



1. Создайте оставшиеся операции, согласно сценарию.

**Задание 4.3. Проектирование диаграммы классов и диаграммы последовательности средствами MS Visio**

Для заданного варианта использования спроектировать:

1. Диаграмму классов, с заданием стереотипов классов и атрибутами классов
2. Диаграмму последовательности с объектами и соответствующими классами. У классов задать атрибуты. У сообщений задать операции.
3. На диаграмме классов каждому классу добавить разработанные в диаграмме последовательности операции.

**Описание предметной области**

В цветочном магазине создается автоматизированная информационная система продаж. Необходимо обеспечить функцию ввода информации о новом растении.

Ввод информации осуществляется из **главной формы** системы. При выборе соответствующей команды открывается **форма ввода растения.** В нее продавец вводит информацию о **растении** (название, условия выращивания, почва), а также устанавливает **сорт** растения.

Сорт выбирается следующим образом: из формы ввода растения путем выбора соответствующей команды открывается **форма справочника сортов растений**, в которую из соответствующей базы данных подгружается информация о всех существующих сортах. Продавец выбирает нужный сорт (например, двойным щелчком мыши) и он заносится в соответствующее поле в форме ввода растения.

После того, как все данные введены, продавец выбирает на форме команду сохранения и введенная информация сохраняется в соответствующих таблицах БД.

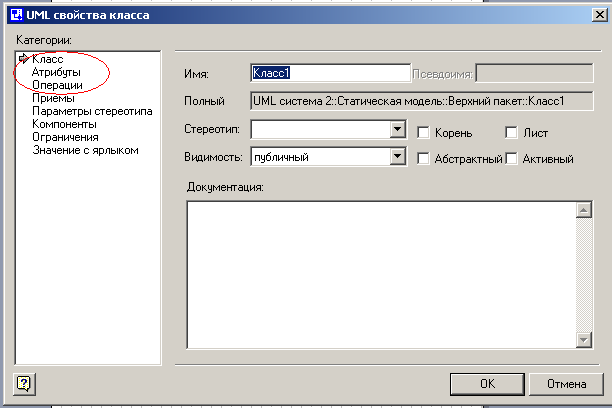
**Методические рекомендации**

**Построение диаграммы классов**

Диаграмма классов строится в разделе Software-UML Model Diagram-UML Static Structure.

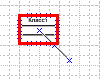
В нашем примере будем использовать образ Class для создания классов-сущностей и образ Interface 

Для задания свойств, атрибутов и операций классу необходимо в рабочем поле дважды щелкнуть по образу класса. В появившемся диалоговом окне ввести необходимую информацию.

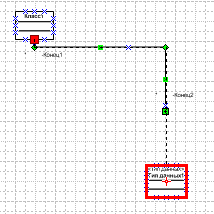


У Интерфейсов атрибуты отсутствуют.

Операции классов и интерфейсов на первом этапе можно не вводить. Они добавляются после проектирования диаграммы последовательности.

Связи добавляются при помощи линии Binary Association. Для грамотного проектирования ассоциации необходимо подвести один конец линии ассоциации к классу таким образом, чтобы его периметр выделился красным цветом. 

Затем установить второй конец на выделенном образе другого класса



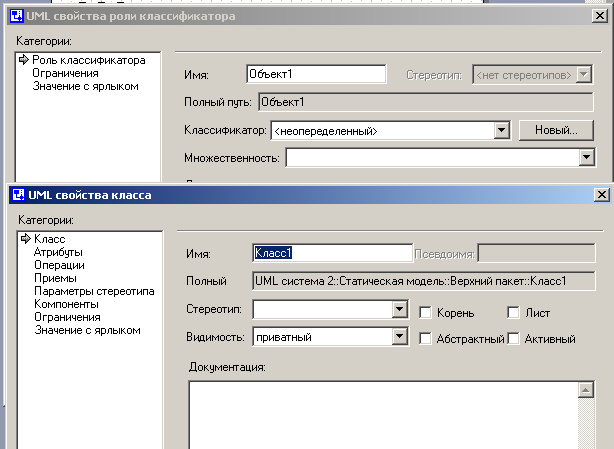
Для редактирования свойств ассоциации необходимо дважды щелкнуть по линии. В появившемся окне Свойства ассоциации при необходимости задать имя и мощность ассоциации.

**Построение диаграммы последовательности**

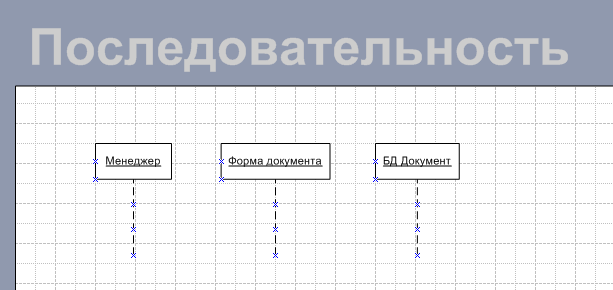
Диаграмма классов строится в разделе Software-UML Model Diagram-UML Sequence.

На диаграмме последовательности отображаются конкретные экземпляры класса – объекты, под которыми проектируется линия жизни объекта . Для редактировании объекта дважды щелкните по нему. В окне «UML cвойства роли классификатора» вводим имя объекта (поле Имя). В поле классификатор указываем класс, к которому принадлежит данный объект. Если такого класса еще не спроектировано, нажимаем кнопку Новый…

В появившемся окне «UML свойства класса» указываем имя класса, устанавливаем атрибуты (если они есть)

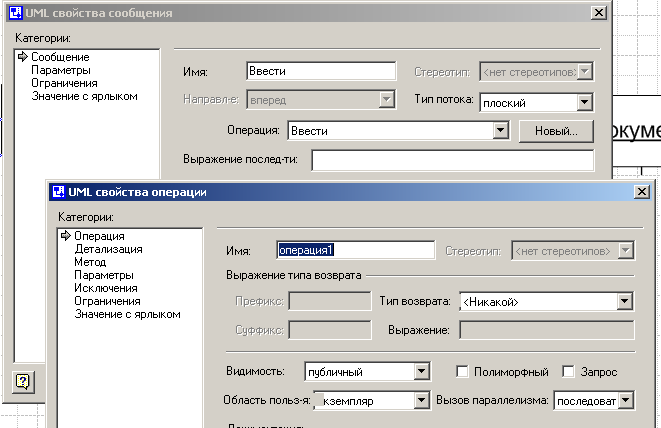


Объект может быть не только экземпляром класса, но и актером. В этом случае в окне «UML свойства класса» в поле Имя указываем **Актер. Примерный**  вид начального этапа проектирования диаграммы последовательности



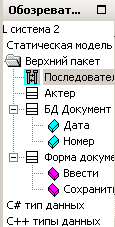
На диаграмме последовательности объекты обмениваются между собой сообщениями 

Для ввода сообщения необходимо дважды щелкнуть по стрелке.



В окне «UML свойства сообщения» вводится имя сообщения. Каждое сообщение соотносится с операциями класса. Для этого, если в выпадающем списке «Операция» присутствует необходимая операция, ее выбирают из списка. Если операция отсутствует, то нажимается кнопка «Новый…» и в открывшемся диалоговом окне «UML свойство операции» вводится новая операция (в поле Имя).

После добавления всех операций и атрибутов классов в поле Обозреватель отображаются атрибуты и операции классов. Например:



## Раздел 5. Технологии создания ПО

**Тема 5.1. Жизненный цикл ПО**

**Задание 5.1** Используя ресурсы Internet найти ГОСТ 12207-2010 «Информационная технология. Системная и программная инженерия. ПРОЦЕССЫ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНЫХ СРЕДСТВ».

Выявите процессы, связанные непосредственно с разработкой модели программного обеспечения ИС.

**Методические рекомендации**

Найти ГОСТ 12207-2010 «Информационная технология. Системная и программная инженерия. ПРОЦЕССЫ ЖИЗНЕННОГО ЦИКЛА ПРОГРАММНЫХ СРЕДСТВ».

Изучить пункт 6 «Процессы жизненного цикла систем». Выявите процессы, связанные непосредственно с разработкой модели программного обеспечения ИС.

Изучить пункт 7 «Процессы жизненного цикла программных систем». Выявите процессы, связанные непосредственно с разработкой модели программного обеспечения ИС.

**Тема 5.2. Технологии создания ПО**

**Цель выполнения задания** - получение практических навыков разработки технического задания на проектирование информационной системы.

**Результатом выполнения** данного заданий является разработанное техническое задание на проектирование информационной системы. Техническое задание формируется в самостоятельном файле Word.

**Методические рекомендации**

1. Изучить содержание ГОСТ 34.602-89. Информационная технология. Автоматизированные системы. Техническое задание на создание автоматизированной системы

2. Создать титульный лист технического задания своего проекта согласно 34.602-89.

3. Второй лист должен содержать оглавление технического задания.

4. Сформировать следующие разделы технического задания (согласно ГОСТ 34.602-89):

1) общие сведения;

2) назначение и цели создания (развития) системы;

3) характеристика объектов автоматизации;

4) требования к системе;

5) состав и содержание работ по созданию системы;

8) требования к документированию;

9) источники разработки.

## Раздел 6. Объектно-ориентированное проектирования программного обеспечения

**Тема 6.1. Моделирование прецедентов**

**Задание 6.1. Проведение объектно-ориентированного анализа проектируемой системы**

**Результатом** выполнения данного заданий является часть проекта информационной системы.

1. Сформулировать функциональные требования к системе.

2. Создать диаграмму вариантов использования.

3. Выполнить подробное описание действующих лиц (актеров) и вариантов использования (в редакторе Word) в виде таблицы.

Варианты предметных областей:

1. Система автоматизации гостиницы
2. Система автоматизации ресторана
3. CRM-система
4. Библиотечная система
5. Система автоматизации автобусного парка
6. Система автоматизации парковки
7. Система автоматизации проката автомобиля
8. Система кредитования банка
9. Система автоматизации турбазы
10. Система автоматизации салона красоты
11. Система автоматизации фитнес-центра

**Методические рекомендации**

Необходимо четко, лаконично и формально описать требования к системе. Например, система должна:

1. Вести учет санаториев
2. Вести учет путевок
3. Вести учет клиентов
4. Создавать новый документ продажи путевок клиенту
5. Изменять документ продажи путевок
6. Удалять документ продажи путевок
7. Организовывать хранение документа продажи потевок в журнале документов продаж (осуществлять просмотр, поиск, сортировку в журнале документов)
8. Формировать договор фирмы с санаторием.
9. Рассчитывать комиссионные за проданные путевки
10. Формировать отчет о продажах путевок

Рекомендуется сначала проанализировать те данные, которые будут храниться в вашей системе постоянно (условно-постоянно). Обычно такими данными являются справочники. Затем необходимо сформировать требования, предъявляемые к хранению и обработке заново возникающей информации или часто изменяющейся (документы).

Затем следует описать обработки данных в системе, осуществление поиска, различных расчетов и пр. Затем следует анализировать формирование отчетов системы. В некоторых случаях возникают и другие действия: авторизация пользователя и пр. Требования к таким возможностям системы тоже необходимо формулировать.

Сформулированные требования в дальнейшем будут основой для создания диаграммы вариантов использования.

На диаграмме вариантов использования каждое действующее лицо (actor) и вариант использования должны сопровождаться описанием. Эти описания должны быть составлены на русском языке. Описание действующего лица должно коротко (в одну-две строки) сообщать о роли данного лица. Описание варианта использования должно включать в себя пояснение. Описания представляют собой текст, введенный в поле Documentation спецификации соответствующего элемента диаграммы. Поместить диаграмму в виде картинки в текстовый документ с проектом системы.

**Тема 6.2. Проектирование пользовательского интерфейса**

**Задание 6.2. Проектирование прототипов информационной системы**

**Методические рекомендации**

Спроектировать прототипы интерфейса ИС (любыми средствами). Рекомендуется проектировать прототип для экранной формы любого документа и экранной формы либо журнала документа, либо главной формы. Должно быть представлено не менее двух прототипов каждой формы (разные цветовые комбинации, расположение меню, кнопок, строк и пр., использование разных шрифтов).  Поместить прототипы в виде картинок в документ с проектом системы.

**Тема 6.3. Проектирование баз данных информационных систем**

**Задание 6.3. Проектирование базы данных информационной системы**

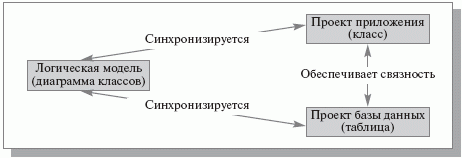
Создать модель предметной области уровня объектно-ориентированного анализа для своей информационной системы Любыми средствами (Word, Excel, Access, Paint и пр.). Поместить модель в виде картинки в текстовый документ с проектом системы.

**Методические рекомендации**

На этом этапе осуществляется отображение элементов полученных ранее моделей классов в элементы моделей базы данных и приложений:

* классы отображаются в таблицы;
* атрибуты – в столбцы;
* типы – в типы данных используемой СУБД;
* ассоциации – в связи между таблицами (ассоциации "многие-ко-многим" преобразуются в ассоциации "один-ко-многим" посредством создания дополнительных таблиц связей);
* приложения – в отдельные классы с окончательно определенными и связанными с данными в базе методами и атрибутами.

Поскольку модели базы данных и приложений строятся на основе единой логической модели, автоматически обеспечивается связность этих проектов.



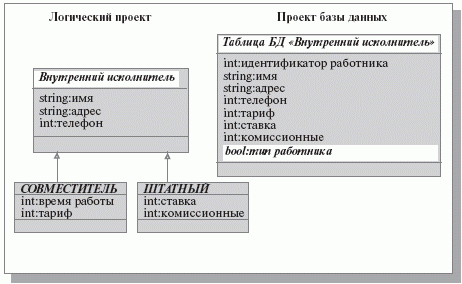
В модель базы данных отображаются только перманентные классы, из которых удаляются атрибуты, не отображаемые в столбцах (например, атрибут типа "Общий объем продаж", который получается суммированием содержимого множества полей базы данных). Некоторые атрибуты (например, АДРЕС) могут отображаться в множество столбцов (СТРАНА, ГОРОД, УЛИЦА, ДОМ, ПОЧТОВЫЙ ИНДЕКС).

Для каждого простого класса в модели базы данных формируется отдельная таблица, включающая столбцы, соответствующие атрибутам класса.

Отображение классов подтипов в таблицы осуществляется одним из стандартных способов:

* одна таблица на класс;
* одна таблица на суперкласс;
* одна таблица на иерархию.

В первом случае для каждого из классов создается отдельная таблица, между которыми затем устанавливаются необходимые связи. Во втором случае создается таблица для суперкласса, а затем в каждую таблицу подклассов включаются столбцы для каждого из атрибутов суперкласса. В третьем – создается единая таблица, содержащая атрибуты как суперкласса, так и всех подклассов. При этом для выделения исходных таблиц подклассов в результирующую таблицу добавляется один или более дополнительных столбцов (на рисунке показан курсивом).



Разработка проекта базы данных осуществляется с использованием специального UML-профиля (Profile for Database Design), который включает следующие основные компоненты диаграмм:

* таблица – набор записей базы данных по определенному объекту;
* столбец – элемент таблицы, содержащий значения одного из атрибутов таблицы;
* первичный ключ (РК) – атрибут, однозначно идентифицирующий строку таблицы;
* внешний ключ (FK) – один или группа атрибутов одной таблицы, которые могут использоваться как первичный ключ другой таблицы;
* обязательная связь – связь между двумя таблицами, при которой дочерняя таблица существует только вместе с родительской;
* необязательная связь – связь между таблицами, при которой каждая из таблиц может существовать независимо от другой;
* представление – виртуальная таблица, которая обладает всеми свойствами обычной таблицы, но не хранится постоянно в базе данных;
* хранимая процедура – функция обработки данных, выполняемая на сервере;
* домен – множество допустимых значений для столбца таблицы.

**Тема 6.4. Проектирование классов**

**Задание 6.4. Провести детальное проектирование выбранной информационной системы**

Провести детальное проектирование выбранной информационной системы, а именно:

1. Спроектировать диаграммы последовательности для тех вариантов использования, для которых сформированы сценарии.

2. Спроектировать классы, участровующие в выбранных вариантах использования, а именно:

2.1. Детализировать проектные классы.

2.2. Уточнить операции классов (Определить полную сигнатуру)

2.3. Уточнить атрибуты классов (тип атрибута)

2.4. Для каждого спроектированного класса в текстовом документе проекта выполнить описание, в котором представить в виде таблицы: название операции/атрибута, сигнатура (описание или тип), назначение.

3. Спроектировать общую диаграмму классов. Уточнить связи между классами на диаграмме.

**Методические рекомендации**

Спроектировать классы для каждого варианта использования, для которых сформированы сценарии. Определить первоначальный набор атрибутов и ассоциаций классов (диаграмма классов для каждого созданного сценария варианта использования уровня предварительного проектирования). Для каждого класса указать стереотип. На диаграмме должны присутствовать классы со стереотипами: граничный, сущность, управляющий. Поместить созданные диаграммы в виде картинок в текстовый документ с описанием проекта.

Спроектировать пакеты (если это необходимо). Пакеты определить самостоятельно, исходя из предметной области и выбранной технологии проектирования. Разместить классы по пакетам. В документ проекта поместить в виде картинки представление логического браузера с размещенными по пакетам классами.

Спроектировать (любыми средствами) модель предметной области уровня предварительного проектирования.

Уточнить операции классов (Определить полную сигнатуру) и уточнить атрибуты классов (тип атрибута).

Для каждого спроектированного класса в текстовом документе проекта выполнить описание, в котором представить в виде таблицы: название операции/атрибута, сигнатура (описание или тип), назначение.

Спроектировать общую диаграмму классов. Уточнить связи между классами на диаграмме.

Поместить модель в отчет в виде картинки.

# Литература для учащегося

**Основные источники:**

1. Вендров А.М. Проектирование программного обеспечения экономических информационных систем: Учебник: М.: Финансы и статистика, 2012.–344 с.
2. Вендров А.М. Практикум по проектированию программного обеспечения экономических информационных систем М.: Финансы и статистика, 2012.– 192 с.
3. Заботина Н.Н., Проектирование информационных систем: Учебное пособие. - М.: НИЦ Инфра-М, 2014. - 331 сURL<http://znanium.com/catalog.php?bookinfo=454282>
4. Н.З. Емельянова, Т.Л. Партыка, И.И. Попов., Проектирование информационных систем: Учебное пособие / - М.: Форум: НИЦ ИНФРА-М, 2014. - 432 с/ URL<http://znanium.com/bookread2.php?book=419815>
5. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений. 3-е изд. – М.: Вильямс, 2010.
6. Якобсон А., Буч Г., Рамбо Дж. Введение в UML от создателей языка. — [ДМК-Пресс](https://www.labirint.ru/pubhouse/1416/), 2015 г.
7. Черемных С.В. и др. Структурный анализ систем: IDEF-технологии / С.В.Черемных, И.О.Семенов, В.С.Ручкин. - М.: Финансы и статистика, 2006.
8. Черемных С.В. и др. Моделирование и анализ систем: IDEF-технологии: практикум / С.В.Черемных, И.О.Семенов, В.С.Ручкин. - М.: Финансы и статистика, 2006.
9. ГОСТ 24.601-86 Автоматизированные системы Стадии создания.
10. ГОСТ 34.601-90 Комплекс стандартов на автоматизированные системы "АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ. СТАДИИ СОЗДАНИЯ"
11. ГОСТ 34. Разработка автоматизированной системы управления (АСУ)

**Дополнительные источники:**

1. Орлов. С. Технологии разработки программного обеспечения: Учебник. - СПб.: Питер, 20012.
2. Иванова Г.С. Технология программирования: Учебник для ВУЗов - М.: Изд-во МГТУ им. Н.Э. Баумана, 2016.
3. Гагарина Л.Г., Кокорева Е.В., Виснадул Б.Д. Технология разработки программного обеспечения: учебное пособие / под ред. Л.Г. Гагариной. – М.: ИД «Форум»: Инфра-М, 2008

**Ресурсы INTERNET**

1. [www.intuit.ru](http://www.intuit.ru) – Интернет-университет информационных технологий
2. vendrov.chat.ru – страница автора учебников [1], [2].
3. [www.it.ru.edu](http://www.it.ru.edu) – Академия IT
4. [www.citforum.ru](http://www.citforum.ru) – центр информационных технологий
5. [www.cetus-links.org](http://www.cetus-links.org) - сборник ссылок по программной инженерии и объектным технологиям
6. [www.bptrends.com](http://www.bptrends.com) – аналитические материалы по моделированию бизнес-процесов
7. [www.interface.ru](http://www.interface.ru) – компания «Интерфейс»
8. [www.rational.com](http://www.rational.com) – компания Rational Software
9. [www.borland.com](http://www.borland.com) – компания Borland
10. [www.cel.cmu.edu](http://www.cel.cmu.edu) – институт программной инженерии (SEI)
11. [www.pmi.org](http://www.pmi.org) – институт управления проектами (PMI)
12. [www.idef.com](http://www.idef.com) – стандарты IDEF
13. [www.cdmagazine.com](http://www.cdmagazine.com) – Журнал Software Development

**Инна Дмитриевна Камскова**

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ по выполнению практических заданий по МДК.01.02 Методы и средства проектирования информационных систе**

Учебно-методическое пособие

Федеральное государственное автономное образовательное  
 учреждение высшего образования

Нижегородский государственный университет имени Н.И. Лобачевского

603950, Нижний Новгород, пр. Гагарина, 23.